
Effiziente, GPU-basierte Simulation thermischer Spritzprozesse

Dissertation

zur Erlangung des Grades eines

D o k t o r s d e r N a t u r w i s s e n s c h a f t e n

der Technischen Universität Dortmund
an der Fakultät für Informatik

von
Thomas Wiederkehr

Dortmund
2015

Tag der mündlichen Prüfung: 1. Oktober 2015

Dekan: Prof. Dr.-Ing. Gernot Fink

Gutachter: Prof. Dr. Heinrich Müller
Prof. Dr. Stefan Turek

Danksagung

An dieser Stelle möchte ich mich bei allen, die mich bei der Entstehung der Dissertation unterstützt haben, bedanken. Insbesondere danke ich Prof. Dr. Heinrich Müller für die Betreuung der Dissertation, seine wertvollen Ratschläge und die Gewährung des notwendigen zeitlichen Freiraums für die Fertigstellung dieser Arbeit. Weiterhin danke ich Prof. Dr. Stefan Turek für die Übernahme der Begutachtung meiner Dissertation.

Den am Sonderforschungsbereich 708 beteiligten Kollegen aus den Fakultäten Informatik, Maschinenbau und Mathematik danke ich für die gute interdisziplinäre Zusammenarbeit und Kollegialität. Insbesondere bedanke ich mich bei Ingor Baumann, Leif Hagen, Benjamin Krebs, Peter Hollingsworth und Mohamed Abdulgader für Ihre vielfältige Unterstützung bei der Realisierung der zahlreichen im Rahmen meiner Arbeit durchgeführten Versuche. Ferner danke ich Carsten Scheele und Benjamin Johnen für ihre Hilfestellungen bei der Robotereinmessung und -steuerung, Achim Peuker für sein Engagement bei der Herstellung der benötigten Proben und Aufspannungen sowie Volker Brandt und Tobias Siebrecht für ihren Einsatz bei der Vermessung zahlreicher Proben und Beschichtungen.

Meinen Kollegen am Lehrstuhl für Graphische Systeme danke ich für das gute Arbeitsklima und die zahlreichen wissenschaftlichen Diskussionen, die mir auch Einblicke in andere Projekte ermöglichten. Mein besonderer Dank gilt Marcel Gaspar, David Fiedler und Denis Fissler für ihre Hilfsbereitschaft und ihre Ratschläge zur Lösung kniffliger Probleme.

Abschließend möchte ich meinen Eltern dafür danken, dass sie es mir ermöglicht haben zu studieren und die Promotion anzustreben. Mein größter Dank gilt jedoch meiner Frau Petra für ihre fortwährende Unterstützung, ihr Verständnis in schwierigen Phasen und dafür, dass sie mich zum Durchhalten motiviert hat.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Beiträge dieser Dissertation	3
1.1.1	Beiträge zur Simulation thermischer Spritzprozesse	4
1.1.2	Beiträge zur Erzeugung, Digit. und Nachbearb. von Footprintprofilen	5
1.2	Eigene Publikationen	6
1.3	Gliederung	8
1.4	Notation	9
1.5	Danksagung	9
2	Grundlagen und Ziele der Arbeit	11
2.1	Thermisches Spritzen	11
2.1.1	Partikelaufrall und Schichtaufbau	12
2.1.2	Industrielle Anwendungsgebiete	14
2.1.3	Lichtbogenspritzen	15
2.2	Maschinenbaulicher Anwendungshintergrund dieser Arbeit	16
2.3	Beschichtungssimulationen	19
2.3.1	Ziele der konzipierten Simulation	20
2.3.2	Verfügbare Simulationssoftware	22
3	Erzeugung, Digitalisierung und Nachbearbeitung von Footprintprofilen	29
3.1	Stand der Forschung und Technik	30
3.2	Übersicht	36
3.3	Probenvorbereitung	39
3.4	Robotergestützte Footprintherstellung	40
3.4.1	Einmessung des Werkstücks	42
3.4.2	Thermische Belastung des Bauteils	51
3.5	Digitalisierung von Footprints	55
3.5.1	Auswahl des Scansystems	55
3.5.2	Scanprozess	60
3.6	Nachbearbeitung dreidimensionaler Profile	62
3.6.1	Ausrichtung der digitalisierten Profile	63
3.6.2	Analyse der unbeschichteten Substrate	65
3.6.3	Korrektur thermischer Verformungen	69

3.6.4	Konvertierung für die Nutzung in der Simulation	73
3.7	Nachbearbeitung zweidimensionaler Profile	78
3.8	Nachbearbeitung durch modellbasierte Optimierung	82
3.8.1	Variation des Footprintprofils	84
3.8.2	Definition der Spritzbahnen	85
3.8.3	Zielfunktion	87
3.8.4	Ergebnisse	88
3.8.5	Ergebnisse zur Effektivität der Korrektur thermischer Verformungen	93
3.9	Funktionsbasierte Footprintrepräsentation	95
3.9.1	Modellfunktionen	96
3.9.2	Eignung der Modellfunktionen	99
3.9.3	Ergebnisse zur funktionsbasierten Profilrepräsentation	104
4	Beschichtungsmodell	109
4.1	Beschichtungsmodelle in der Literatur	110
4.2	Massenstromdichteverteilung	112
4.2.1	Physikalische Modellierung als volumetrische Strömung	112
4.2.2	Footprintexperimente	115
4.2.3	Modellannahmen	116
4.3	Geometrische Übertragungsfunktion	118
4.3.1	Kegelförmiger Spritzstrahl	118
4.3.2	Veranschaulichung der kegelbasierten Übertragungsfunktion	121
4.3.3	Vereinfachte kegelbasierte Übertragungsfunktion	123
4.3.4	Zylindrischer Spritzstrahl	127
4.4	Haftwirkungsgrad	128
4.4.1	Modellierung des Haftwirkungsgrades	129
4.4.2	Experimentelle Bestimmung des Haftwirkungsgrades	132
4.4.3	Automatische Kalibrierung	134
4.4.4	Haftwirkungsgrad für den verwendeten Lichtbogenspritzprozess	135
5	Beschichtungssimulation	137
5.1	OpenGL, GLSL und GPGPU	138
5.1.1	OpenGL und GLSL	139
5.1.2	GPGPU	142
5.2	Globaler Simulationsablauf	143
5.3	Eingabedaten und Geometrirepräsentation	145
5.3.1	Bauteilgeometrie	145
5.3.2	Roboterbahn	147
5.3.3	Prozesskonfiguration	148
5.4	GPU-basiertes Simulationskonzept	149
5.4.1	Analyse der Problemstruktur	149

5.4.2	Konzept zur GPU-beschleunigten Sichtbarkeitsberechnung	152
5.4.3	Minimierung von Speichertransfers	154
5.4.4	Vollständig GPU-basierte Simulation	156
5.5	Implementierung	158
5.5.1	Berechnung oberflächenbezogener Zielgrößen	158
5.5.2	Berechnung bahnbezogener Zielgrößen	171
5.5.3	Optimierung der CPU-seitigen Implementierung	174
5.6	Einbindung einer thermodynamischen Simulation	180
5.7	Visualisierung	180
5.8	Kalibrier- und Analysemodule	181
5.8.1	Automatische Parameterkalibrierung	182
5.8.2	Sensitivitätsanalyse	184
6	Evaluation	185
6.1	Demonstration und Verifikation grundlegender Zusammenhänge	186
6.1.1	Virtueller Footprintversuch	186
6.1.2	Bahnzielgrößen am Beispiel geneigter Ebenen	189
6.1.3	Sichtbarkeitsberechnung und Wahl der Tiefenpufferauflösung	191
6.2	Simulationsgeschwindigkeit	195
6.2.1	Laufzeitmessung und Analyse	196
6.2.2	Vergleich mit Nvidia Optix	199
6.3	Beschichtung von Tiefziehwerkzeugen	202
6.3.1	Bauteile und Versuchsdurchführung	203
6.3.2	Messung der Schichtdickenverteilung	204
6.3.3	Simulationsergebnisse	211
6.4	Sensitivitätsanalyse und simulationsgestützte Bahnoptimierung	221
6.4.1	Sensitivitätsanalyse für Tiefziehwerkzeuge	221
6.4.2	Einsatz der Simulation zur Bahnoptimierung	223
7	Zusammenfassung	227
	Anhang	231
A	Flächeninhalt eines elliptischen Kegelschnitts	231
B	GLSL-Methoden zur Kubikwurzelberechnung	235
C	Shadercode zur Berechnung bahnbezogener Zielgrößen	237
D	Beschichtete Matritzenbauteile	241
	Abbildungsverzeichnis	247
	Tabellenverzeichnis	250
	Literaturverzeichnis	251

Einleitung

Computergestützte Simulationen technischer Prozesse finden sich heute in einer Vielzahl industrieller Fertigungsbereiche und dienen der Planung und Optimierung qualitativer und ökonomischer Aspekte von Produktionsabläufen. Die simulative Vorhersage von Eigenschaften des zu fertigenden Produktes in Abhängigkeit von Parametern des Prozesses ermöglicht in vielen Fällen eine erhebliche Kostenersparnis durch die Einsparung von Trial-and-Error-Schleifen in der Bauteilentwicklung oder die Optimierung des Prozesses im Hinblick auf Material-, Energie- oder Zeitersparnis. Ebenso kann eine Optimierung der Prozessparametrisierung oder der Eingriffsbedingungen zu einer Steigerung der Qualität gefertigter Bauteile führen.

Diese Arbeit befasst sich mit der Konzeption und Realisierung eines Simulationssystems für thermische Spritzprozesse. Thermische Spritzverfahren stellen eine Gruppe von Verfahren zur Beschichtung von Bauteilen mit verschiedensten Werkstoffen dar, bei denen der Schichtwerkstoff in einem Spritzbrenner durch die Zuführung thermischer und kinetischer Energie zunächst geschmolzen und anschließend mittels eines Zerstäubergases in Form geschmolzener Partikel in Richtung der zu beschichtenden Oberfläche beschleunigt wird. Beim Aufprall auf die kältere Bauteiloberfläche kühlen die Partikel schlagartig ab, erstarren und formen dabei durch Verschmelzen und mechanische Verklammerung mit dem Substrat eine Beschichtung, welche typischerweise Schichtdicken zwischen $50\ \mu\text{m}$ und $1\ \text{mm}$ erreicht [85]. Die Vielzahl verwendbarer Schichtwerkstoffe, welche Metalle, Legierungen, Keramiken und Polymere umfassen, ermöglicht die Herstellung von Beschichtungen mit verschiedensten funktionalen Charakteristiken, welche die Eigenschaften des Grundwerkstoffes verbessern oder neue hinzufügen. Hierzu zählen mechanische, elektrische, chemische, thermophysikalische, magnetische oder optische Eigenschaften [38, 115]. Beispiele sind thermisch isolierende Beschichtungen (engl.: Thermal Barrier Coating, TBC) für Schaufeln von Turbinen und Strahltriebwerken,

Korrosionsschutzbeschichtungen in der Schifffahrts- und Automobilindustrie oder Verschleißschutzbeschichtungen für Umformwerkzeuge, welche Gegenstand der Forschung im Sonderforschungsbereich 708 [141–144, 151, 152] darstellen und im Rahmen dieser Arbeit als Fertigungsbeispiele dienen.

Je nach Einsatzzweck ergeben sich unterschiedliche Anforderungen an die Beschichtung. Die in vielen modernen Anwendungsgebieten thermisch gespritzter Beschichtungen wichtigste Anforderung stellt dabei das Erreichen einer geforderten Schichtdickenverteilung innerhalb enger Maßtoleranzen dar, welche entscheidend für die Leistung und die Lebensdauer der gefertigten Komponenten ist [121]. Dies trifft auf die genannten thermischen Isolationsbeschichtungen und Verschleißschutzschichten für Umformwerkzeuge sowie beispielsweise auf die Beschichtung von Zylinderbohrungen, Kurbelgehäusen und Walzen für die Papierindustrie zu [38, 80, 97, 111]. Weitere Anforderungen leiten sich aus unterschiedlichen mechanischen Eigenschaften wie Schichthaftung, Härte, Porosität oder Rauheit sowie chemischen Eigenschaften wie Oxidgehalt und Phasenzusammensetzung ab. Abgesehen von der Wahl der Schichtwerkstoffe werden viele dieser Eigenschaften durch die Führung des Spritzbrenners über die zu beschichtende Oberfläche, die dabei auftretenden geometrischen Eingriffsbedingungen sowie die daraus resultierende thermische Belastungshistorie des Bauteils bestimmt oder beeinflusst. Der Fertigungsplanung kommt daher eine große Bedeutung zu.

Für einfache großflächige oder rotationssymmetrische Bauteile kann der Spritzbrenner händisch oder mittels einer Linearverfahreinheit über die Bauteiloberfläche geführt werden. Die zunehmende Anzahl zu beschichtender Bauteile mit komplexen Geometrien erfordert jedoch den Einsatz von Robotern zur Führung des Spritzbrenners [85], um eine den Anforderungen entsprechende und über mehrere Bauteile hinweg reproduzierbare Schichtverteilung zu erzeugen. Der Entwurf und die Programmierung der hierfür benötigten Roboterbahn erfolgt dabei in den meisten Fällen durch eine Online-Programmierung des Roboters in der Spritzkabine [41, 43, 59, 85, 121], bei der der Roboter mittels einer Steuereinheit manuell von Punkt zu Punkt auf dem Bauteil verfahren und die entstehende Bahn abgespeichert wird. Viele Vorteile bieten demgegenüber Offline-Bahnplanungsalgorithmen zur automatischen Berechnung von Bahnverläufen auf Basis von CAD-Daten der Bauteilgeometrie. So sind die eingesetzten Planungsalgorithmen schneller und genauer als Online-Planungsmethoden, blockieren keine Produktionseinrichtungen und besitzen das Potential, neben der reinen Berechnung paralleler Bahnkurven mit vorgegebenem Abstand in der Bauteiloberfläche auch roboter- und prozessspezifische Randbedingungen im Rahmen einer simulationsgestützten Bahnplanung oder -optimierung zu berücksichtigen [20]. Beispielsweise schätzen [128] die Ersparnis durch Offline-Bahnplanungsmethoden für die Beschichtung einer Turbinenschaufel auf 50.000 bis 100.000 US\$.

Ein weit verbreiteter Ansatz zur Offline-Bahnplanung besteht darin, eine Schar quasi-paralleler Kurven auf der zu beschichtenden Oberfläche zu erstellen und diese außerhalb der Fläche zu einer Mäander- oder Spiralbahn zu verbinden [81, 108], wobei in vielen Fällen als einzelner zentraler geometrischer Steuerparameter der von der Beschichtung ebener oder zylindrischer Bauteile bekannte Bahnoffset Verwendung findet. Für eine umfassende, anforderungsgerech-

te Prozessauslegung genügt diese einfache Übertragung der von ebenen Bauteilen bekannten Mäanderbahnstrategie auf komplexe Bauteile jedoch nicht [41, 121], da diese weder die wechselnden geometrischen Eingriffsbedingungen, wie beispielsweise Spritzabstände oder -winkel, noch das dynamische Verhalten des Roboters einbezieht. Insbesondere können derartige, rein geometrische Bahnplanungsansätze keine anforderungsgerechten und offline berechneten Lösungen bieten, welche unumgängliche lokale Abweichungen wichtiger Eingriffsparameter, wie Verfahrgeschwindigkeit, Abstand oder Anstellwinkel des Spritzbrenners, kompensieren. Auf dem Gebiet der Bahnplanung existieren in der aktuellen Literatur bereits Ansätze zur Simulation und Analyse der Roboterkinematik und -dynamik [43, 52] sowie zur Vorhersage der Temperaturentwicklung im Bauteil [15, 70]. Die Einstellung der gewünschten Schichtdickenverteilung erfolgt jedoch in der Regel lediglich indirekt durch das Bestreben, eine konstante Verfahrgeschwindigkeit des Spritzbrenners einzuhalten. Eine direktere, simulationsbasierte Berücksichtigung der abgelagerten Schichtdickenverteilung in der Offline-Bahnplanung wäre an dieser Stelle von großem Vorteil.

Im Kontext der Modellierung des Schichtablagerungsprozesses sind die wesentlichen Zusammenhänge zwischen den geometrischen Eingriffsparametern und der abgelagerten Schichtdicke, nicht zuletzt durch eine Vielzahl von Versuchsreihen für unterschiedliche Spritzverfahren, bekannt [85, 91, 150, 159, 179, 180]. Eine Reihe von Veröffentlichungen der letzten Jahre stellen mathematisch-algorithmische Beschichtungsmodelle auf, welche diese Zusammenhänge beschreiben. Die meisten dieser Modelle betrachten jedoch spezifische Anwendungsszenarien und eignen sich daher nicht für die Beschichtungsberechnung im allgemeinen Fall komplexer, dreidimensionaler Bauteile und beliebiger Roboterbahnen [121]. So sind einige dieser Modelle zweidimensional [46, 158], andere sind beschränkt auf einen konstanten Spritzabstand oder Spritzwinkel [15], rotationssymmetrische Massenflusscharakteristiken [34, 44] oder sogar auf bestimmte Bauteil- oder Bahnformen [15, 34, 54]. Ein Grund für die Beschränkung zahlreicher Modelle und veröffentlichter Berechnungsbeispiele auf die Verwendung einfachster Bauteile und Verfahrenswege stellt die dadurch vermeidbare oder deutlich vereinfachte algorithmische Komplexität der Sichtbarkeitsberechnung zur Ermittlung der von einer Brennerposition zu beschichtenden Oberflächenteile dar. Insbesondere in Veröffentlichungen zur Simulation des mit dem thermischen Spritzen verwandten Sprühkompaktierens, in dem – aufgrund der sehr dicken abgelagerten Schichten – nicht auf die Sichtbarkeitsberechnung verzichtet werden kann, werden häufig einfache Verfahren zur Sichtbarkeitsberechnung eingesetzt oder vereinfachende geometrischen Annahmen getroffen [34, 69].

1.1 Beiträge dieser Dissertation

Im Rahmen dieser Dissertation wurde der aus diesen Betrachtungen abgeleitete Bedarf umfassenderer effizienter Simulationsmethodiken aufgegriffen und ein GPU-basiertes Softwareframework zur Simulation der Beschichtung komplexer Bauteile in thermischen Spritzprozessen konzipiert und realisiert. Zentrale Ziele, die mit dieser Entwicklung erreicht wurden, sind die

Unterstützung beliebig komplexer Bauteile und Roboterbahnen, die Fähigkeit, unterschiedliche Spritzverfahren abbilden zu können, sowie die effiziente Einsetzbarkeit der Beschichtungssimulation zur automatisierten Offline-Bahnplanung und Fertigungsoptimierung.

1.1.1 Beiträge zur Simulation thermischer Spritzprozesse

Im Folgenden werden zunächst die Beiträge dieser Arbeit zur Simulation thermischer Spritzprozesse aufgeführt.

Footprint-basiertes Beschichtungsmodell. Zur Abbildung der Spritzcharakteristiken unterschiedlicher Verfahren wird in der präsentierten Simulation die Abbildung thermischer Spritzprozesse mittels eines neuartigen, im Rahmen dieser Arbeit entwickelten Footprint-basierten Beschichtungsmodells realisiert, welches die Verwendung beliebiger Massenflusscharakteristiken zulässt. Insbesondere wird dabei weder ein rotationssymmetrischer, noch ein gaußförmiger Massenfluss vorausgesetzt. Stattdessen erfolgt die simulative Abbildung der Charakteristik eines Prozesses mittels dreidimensionaler Footprintprofile, welche die von einem stationärem Spritzbrenner unter senkrechtem Anstellwinkel auf einer ebenen Oberfläche abgelagerten Schichtprofile darstellen. Die Simulation unterschiedlicher Spritzprozesse wird auf einfache Weise durch den Austausch des verwendeten Footprintprofils ermöglicht.

GPU-beschleunigte Beschichtungssimulation. Zusätzlich zur Unterstützung beliebig komplexer dreidimensionaler Bauteile sowie beliebiger Roboterbahnen ist das Erreichen einer hohen Simulationsgeschwindigkeit notwendig, um die entwickelte Beschichtungssimulation im Rahmen einer automatisierten Offline-Bahnplanung gewinnbringend einsetzen zu können. Dies ist darin begründet, dass die Beschichtungssimulation im Rahmen eines solchen Planungsprozesses typischerweise vielfach mit modifizierten Bahnen ausgeführt werden muss, um eingebrachte Verbesserungen zu bewerten oder numerische Gradienten in einem Optimierungsverfahren zu berechnen. Diese Zielsetzung wurde in dem in dieser Dissertation vorgestellten Simulationsframework durch die GPU-basierte Implementierung erreicht, welche die im Vergleich zu CPUs deutlich höhere parallele Rechenleistung moderner Grafikkarten nutzt, um für die vorgestellten komplexen Bauteile Simulationszeiten im unteren einstelligen Sekundenbereich zu realisieren. Im Mittelpunkt steht dabei die Adaption der klassischen Renderingpipeline für die Simulation des Schichtauftrags. Die Effizienz dieses Ansatzes wird im synthetischen Benchmarkvergleich der erstellten Implementierung mit dem CUDA-basierten Framework „Optix Prime“ der Firma Nvidia sowie anhand von Simulationsbeispielen deutlich aufgezeigt.

Explizite Unterstützung der Fertigungsplanung. Mit der zusätzlichen Analyse der Prozesseffizienz und der Berechnung geometrischer Eingriffsparameter für jede Bahnposition des Spritzbrenners wird eine intuitive Identifikation kritischer Bahnabschnitte im Rahmen der Fertigungsplanung ermöglicht. Durch die Realisierung eines Netzwerkinterfaces werden automatisierte Bahnplanungs- und -optimierungssysteme explizit unterstützt. Die visuelle Aufbe-

reitung und Darstellung aller Simulationsergebnisse vereinfacht darüber hinaus die fachliche Beurteilung eines Fertigungsaufbaus.

1.1.2 Beiträge zur Erzeugung, Digitalisierung und Nachbearbeitung von Footprintprofilen

Neben der Konzeption des Footprint-basierten Beschichtungsmodells und seiner Implementierung im Rahmen der Beschichtungssimulation wird die experimentelle Herstellung, Digitalisierung und Nachbearbeitung der zugrunde liegenden Footprintprofile in dieser Arbeit ausführlich thematisiert. In diesem Zusammenhang werden insbesondere potentielle technische Schwierigkeiten bei der Herstellung und Vermessung von Footprintproben identifiziert, analysiert und technische sowie algorithmische Lösungen für verschiedene Einzelprobleme dargestellt, um die Zuverlässigkeit und Genauigkeit der Profilakquisition und der Beschichtungssimulation zu steigern.

Lasergestützte Einmessvorrichtung. Die grundlegende Bauweise des im Rahmen dieser Arbeit eingesetzten Lichtbogenspritzbrenners lässt keine exakte Bestimmung der Spritzrichtung zu und die unzureichende Steifigkeit der Brennerhalterung erschwert eine wiederholbare, robotergestützte Positionierung relativ zu einem zu beschichtenden Bauteil. Eine wesentliche technische Neuerung stellt in diesem Kontext die im Rahmen dieser Arbeit entwickelte, direkt an der Düse des verwendeten Lichtbogenspritzbrenners montierbare Lasereinmessvorrichtung dar. Diese erlaubt die Erkennung und Kompensation translatorischer und rotatorischer Abweichungen aller Achsen des Brenners und ermöglicht, unabhängig von den genannten Problemen, eine Einmessung des Brenners auf eine zu beschichtende Bauteilgeometrie.

Vergleich verschiedener Scansysteme. Zur Digitalisierung von Footprint- und Schichtprofilen wurden in dieser Arbeit sechs 3D-Scanner unterschiedlicher Bau- und Funktionsweise auf ihre Eignung zur flächigen Schichtdickenmessung untersucht und untereinander sowie mit der etablierten und sehr genauen, aber zweidimensionalen Messmethodik der lichtmikroskopischen Schichtdickenmessung verglichen.

Schichtvermessung in metallographischen Querschliffen. Zu den algorithmischen Beiträgen dieser Arbeit im Kontext der allgemeinen und Footprint-spezifischen Profilvermessung zählt der Entwurf eines Algorithmus zur automatisierten Vermessung von Schichtprofilen in lichtmikroskopischen Querschliffaufnahmen mit Methoden der digitalen Bildverarbeitung, welcher als eigenständiges Softwaretool realisiert wurde. Im Vergleich zur herkömmlichen Platzierung einzelner Messpunkte am Mikroskoparbeitsplatz erlaubt die entwickelte Methodik eine hoch aufgelöste Abtastung des Profils in einem Bruchteil der Zeit. Durch die Unabhängigkeit der Software von manuell platzierten Messpunkten wird eine objektivere Messung ermöglicht. Die Software wurde ebenfalls im Kontext anderer Arbeiten am Lehrstuhl für Werkstofftechnologie der TU Dortmund für verschiedene Messreihen erfolgreich eingesetzt.

Kompensation thermischer Verformungen. Ein wesentliches Problem bei der extrinsischen taktilen oder optischen Vermessung von Footprintprofilen stellt die mit der hohen thermischen Belastung im Fertigungsprozess einhergehende Verformung des Schicht-Substrat-Verbunds dar. Um eine großflächige extrinsische Vermessung des Schichtprofils trotz des unbekanntem Verlaufs der Substratoberfläche unterhalb der Beschichtung zu ermöglichen, werden im Rahmen dieser Arbeit zwei neue Methoden zur Schätzung und zur algorithmischen Kompensation der Substratverformung vorgestellt und miteinander verglichen.

Modellbasierte Optimierung von Footprintprofilen. Des Weiteren wird in dieser Arbeit ein neuartiger Ansatz zur modellbasierten Optimierung von Footprintprofilen auf Basis von Querschliffmessungen in Linearbahnprofilen vorgestellt. Aufgrund der hohen Genauigkeit der mikroskopiebasierten Messmethodik für die Querschliffprofile und der realitätsnahen Prozessbedingungen bei der Fertigung der zur Optimierung verwendeten Linearbahnprofile kann mit diesem Verfahren eine Steigerung der Profilgenauigkeit erreicht werden.

1.2 Eigene Publikationen

Wesentliche Teile der im vorherigen Abschnitt beschriebenen Beiträge dieser Arbeit sind in einer Reihe von Publikationen des Autors in Fachzeitschriften und auf internationalen Tagungen veröffentlicht worden. Im Folgenden werden diese Publikationen thematisch gegliedert aufgeführt und die Beiträge des Autors zu den einzelnen Veröffentlichungen präzisiert.

Konzeption des Beschichtungsmodells und der Simulation

Die Beiträge zur GPU-basierten Beschichtungssimulation und zu dem entworfenen Footprint-basierten Beschichtungsmodell wurden in drei begutachteten Konferenzbeiträgen veröffentlicht [170, 173, 174]. In dieser Arbeit finden sich diese Themen in den Kapiteln 4 und 5. Die erste Veröffentlichung [170] befasst sich dabei noch mit einer frühen Version der in dieser Dissertation präsentierten Beschichtungssimulation, welche das grundlegende Konzept der GPU-basierten Simulationsmethodik für komplexe Bauteile beschreibt. Die Ideen, die Umsetzung und die Ausführungen zu dieser Methodik wurden vom Autor entwickelt. Die Beiträge der Koautoren beschränken sich hierbei auf technische Diskussionen zu Einzelaspekten mit dem Betreuer, Prof. Dr. Müller, sowie die Bereitstellung von Roboterbahnen für die durchgeführten Beispielsimulationen und Visualisierungen für die Veröffentlichung (A. Kout).

Die beiden weiteren Veröffentlichungen [173, 174] zu diesem Thema wurden auf der International Thermal Spray Conference & Exposition (ITSC) 2009 und 2015 vorgestellt. Gegenüber der erstgenannten Veröffentlichung, in der die Schichtbildung mit dem Modell von Duncan et al. [48] berechnet wurde, werden in [174] das vom Autor entwickelte Footprint-basierte Beschichtungsmodell sowie Ergebnisse von Validierungsversuchen auf ebenen Geometrien vorgestellt. Auch die Fertigung und Digitalisierung der benötigten Footprintprofile wird hier bereits thematisiert. In der aktuellsten Publikation [173] zu diesem Themenbereich wird der finale Entwicklungsstand der Beschichtungssimulation unter Einbeziehung der Shader-basierten

Implementierung präsentiert. Darüber hinaus werden in diesem Beitrag Ergebnisse von Validierungsexperimenten auf Basis komplexer Bauteile sowie Resultate einer automatisierten Bahnoptimierung präsentiert. Für beide Veröffentlichungen, [174] und [173], beschränken sich die Beiträge der Koautoren vom Lehrstuhl für Werkstofftechnologie – im einzelnen M. Abdulgader, B. Krebs, L. Hagen und W. Tillmann – auf die Unterstützung der Durchführung von Spritzexperimenten zur Fertigung von Footprintprofilen und Validierungsversuchen an einfachen und komplexen Geometrien. Die Konzeption und Auswertung der Versuche wurden vom Autor dieser Arbeit durchgeführt. Für die Veröffentlichung [173] stellte D. Hegels ferner die von ihm entwickelte Bahnoptimierungssoftware, welche gemeinsam mit dem Autor dieser Dissertation an die hier präsentierte Beschichtungssimulation angebunden wurde, für die beispielhafte Durchführung von Bahnoptimierungen zur Verfügung. Die in beiden Veröffentlichungen präsentierten Ideen, deren Umsetzung sowie die Ausführungen selbst stammen vom Autor dieser Dissertation, wobei der Betreuer der Arbeit beratend zur Seite stand.

Konzepte zur Footprinterfassung

In dem Themenbereich der Erzeugung, Digitalisierung und Nachbearbeitung von Footprintprofilen, welcher in dieser Arbeit in Kapitel 3 thematisiert wird, entstand die Publikation [171]. Die darin diskutierten Problemstellungen, die vorgeschlagene Lösungsmethodik sowie die Ausarbeitung basieren auf den Ideen und Ausführungen des Autors. Mathematisch-algorithmische sowie konzeptionelle Details wurden gemeinsam mit dem Koautor verfeinert.

Anwendung der Beschichtungssimulation

Eigene Publikationen, in denen die Anwendung der entwickelten Beschichtungssimulation im Vordergrund steht, umfassen zwei Arbeiten zur simulationsbasierten Bahnoptimierung [71, 148] und einen Beitrag zur Simulation der Temperaturentwicklung in komplexen Bauteilen im Beschichtungsprozess [10]. Die Veröffentlichung [71] präsentiert ein Verfahren zur Simulations-gestützten Optimierung von Roboterbahnen im Hinblick auf die erzielte Schichtdickenverteilung auf komplexen Bauteiloberflächen. In [148] wird dieses Verfahren unter zusätzlicher Betrachtung der Roboterdynamik zur Optimierung der Beschichtung eines konkreten Bauteils mit dem HVOF-Verfahren (High Velocity Oxygen Fuel spraying) angewendet. Die Beiträge des Autors zu diesen Veröffentlichungen sind die Realisierung der programmatischen Anbindung der entwickelten Beschichtungssimulation an das Optimierungssystem, die Beteiligung an der Konzeption des Gesamtsystems sowie die Ausführungen zur Beschreibung der Beschichtungssimulation. Die Komponenten zur Simulation der Roboterdynamik wurden am Institut für Produktionssysteme (IPS) entwickelt und die experimentelle Durchführung und Auswertung fand am Lehrstuhl für Werkstofftechnologie (LWT) statt.

Die Veröffentlichung [10] zur Simulation der Temperaturentwicklung im Bauteil entstand in enger Kooperation mit dem Institut für Mechanik der TU Dortmund. Die Beiträge zu dieser Arbeit entfallen zirka je zur Hälfte auf die beteiligten Institute. Dabei wurde das vorgestellte Finite-Elemente-Modell zur Temperatursimulation prototypisch von den Koautoren des Insti-

tuts für Mechanik entworfen und vom Autor dieser Dissertation in die Programmiersprache C++ übersetzt, parallelisiert und in die Beschichtungssimulation integriert.

Anwendung der Konzepte zur Footprinterfassung

In den Publikationen [146, 147, 149] konnte der Autor seine Expertise im Bereich der Footprinterzeugung und -vermessung für den Lichtbogenspritzprozess und das HVOF-Verfahren einbringen. Im Vordergrund stand bei dieser Mitwirkung die Anwendung der Methoden aus Kapitel 3 dieser Arbeit. Das Kernforschungsgebiet liegt bei diesen Veröffentlichungen jedoch im materialwissenschaftlichen Bereich und die Ausführungen sind dementsprechend primär den Koautoren am Lehrstuhl für Werkstofftechnologie zuzuschreiben.

Weitere Publikationen

Vier weitere Publikationen [83, 168, 169, 172] des Autors befassen sich mit der Konzeption effizienter Simulationsansätze für Problemstellungen aus dem Bereich des thermischen Spritzens. Diese thematisieren die Analyse von Schichtmikrostrukturen [168], die Simulation des mikrostrukturellen Schichtaufbaus auf Basis von Einzelpartikelablagerungen [83, 172] sowie die Simulation von Eingriffskräften beim Verdichten poröser thermisch gespritzter Schichten durch einen Kugelwalzprozess [169]. Diese Problemstellungen und Beiträge sind jedoch nicht Gegenstand des wissenschaftlichen Beitrags dieser Arbeit.

1.3 Gliederung

Das folgende Kapitel 2 beschreibt zunächst die Grundlagen thermischer Spritzprozesse, insofern sie für das Verständnis der weiteren Ausarbeitungen relevant sind. Der Fokus liegt dabei auf dem im Rahmen dieser Arbeit verwendeten Lichtbogenspritzprozess. Des Weiteren werden in diesem Kapitel vorhandene kommerzielle Simulationssysteme für thermische Spritzprozesse betrachtet und die genauen Ziele, die mit der Konzeption der in dieser Arbeit vorgestellten Beschichtungssimulation verfolgt wurden, werden definiert.

Kapitel 3 beschreibt die chronologische Vorgehensweise zur Erzeugung, Digitalisierung und Nachbearbeitung von Footprintprofilen für die Verwendung in der Beschichtungssimulation. Dabei wird umfassend auf die auftretenden fertigungsbedingten und messtechnischen Probleme und die im Rahmen dieser Dissertation erarbeiteten Lösungen eingegangen. Im darauffolgenden Kapitel 4 wird das Footprint-basierte Beschichtungsmodell für kegelförmige und zylindrische Spritzstrahlgeometrien beschrieben und veranschaulicht. Kapitel 5 befasst sich schließlich mit der Beschreibung des implementierten Simulationsframeworks sowie insbesondere der Konzeption der GPU-beschleunigten Beschichtungssimulation, in der das Footprint-basierte Beschichtungsmodell zum Einsatz kommt. Darin werden sowohl das grundlegende, auf der OpenGL-Grafikpipeline und GLSL-Shaderprogrammen basierende Simulationskonzept sowie einige für die Simulationsgeschwindigkeit relevante Implementierungsdetails er-

läutert. Die vorgestellte Simulation wird schließlich in Kapitel 6 in Bezug auf die erreichte Simulationsgeschwindigkeit und die Genauigkeit der berechneten Schichtdickenverteilung für komplexe Bauteile evaluiert. In diesem Kapitel wird auch der Einsatz der Beschichtungssimulation im Rahmen der Roboterbahnoptimierung demonstriert. Kapitel 7 fasst die Dissertation kurz zusammen.

1.4 Notation

Die in dieser Dissertation verwendete Notation orientiert sich in den verschiedenen Kapiteln an der typischen Notation der jeweiligen Themenbereiche und ist daher in Bezug auf die verwendeten Symbole nicht übergreifend beschreibbar. Die lokal verwendeten Variablen sind stets an den entsprechenden Stellen beschrieben und erschließen sich häufig als Kürzel der von ihnen bezeichneten Größen. Übergreifend werden in der Regel Skalare als Kleinbuchstaben a, b, \dots , (Spalten-)Vektoren als fettgedruckte Kleinbuchstaben $\mathbf{a}, \mathbf{b}, \dots$, Matrizen als fettgedruckte Großbuchstaben $\mathbf{A}, \mathbf{B}, \dots$ und Mengen als Großbuchstaben A, B, \dots dargestellt. Insbesondere im Kontext der Beschreibung von gemessenen oder simulierten Schichtprofilen werden Punktmengen ferner mit kalligraphischen Großbuchstaben $\mathcal{F}, \mathcal{L}, \mathcal{P}, \dots$ dargestellt. Winkel werden mit griechischen Kleinbuchstaben α, β, \dots bezeichnet.

Für Vektoren geeigneter Dimension bezeichnen $|\mathbf{a}|$ die euklidische Norm, $\langle \mathbf{a}, \mathbf{b} \rangle$ das Skalarprodukt sowie $\sphericalangle(\mathbf{a}, \mathbf{b})$ den eingeschlossenen Winkel. Für Mengen bezeichnet $|A|$ die Kardinalität von A .

In Indizes auftretende Notationen der Form $x_{a|b}$ bedeuten die Gültigkeit der Gleichung für x_a und x_b , ein Sternchen x_* als Index zeigt die Gültigkeit für alle in dem jeweiligen Kontext definierten Indizes von x an. Die besonders häufig auftretenden Indizes \bullet_{sim} und \bullet_{real} kennzeichnen simulierte respektive experimentell ermittelte Größen. Insbesondere im Rahmen des in Kapitel 4 vorgestellten Beschichtungsmodells zeigen Indizes $\bullet_{\mathcal{F}}$ und $\bullet_{\mathcal{W}}$ geometrische Größen der Footprintgeometrie bzw. der Werkstückgeometrie an.

Im Kontext der Beschreibung von Quellcode werden Variablen- und Funktionsnamen im Fließtext in der Form `glReadPixels` gekennzeichnet. Bei Kurzverweisen auf in dem jeweiligen lokalen Kontext nicht verwendete oder nicht näher beschriebene Bezeichner wird ein Pfeil vorangestellt (\rightarrow `glViewport`).

1.5 Danksagung

Diese Dissertation ist im Rahmen des Sonderforschungsbereichs 708, TP B1 entstanden. Der Autor dankt der Deutschen Forschungsgemeinschaft für die finanzielle Förderung.

Grundlagen und Ziele der Arbeit

In diesem einführenden Kapitel werden zunächst thermische Spritzprozesse im Allgemeinen sowie insbesondere das Verfahren des Lichtbogensspritzens grundlegend erläutert. Letzteres wird im Rahmen des Sonderforschungsbereichs 708, in dem auch diese Arbeit entstanden ist, für die Beschichtung von Tiefziehwerkzeugen eingesetzt und dient als Basis für sämtliche in dieser Dissertation durchgeführten Beschichtungsversuche. Die Ausführungen beschränken sich dabei auf die zum Verständnis dieser Arbeit erforderlichen Grundlagen und bieten einen Querschnittseinblick in das übergreifende Thema der Oberflächentechnik. Für einen tiefgreifenderen Einblick in dieses Thema werden die Standardwerke [8, 11, 38, 115] empfohlen. Im Anschluss an diese Einführung wird in Kapitel 2.2 auf den maschinenbaulichen Hintergrund dieser Arbeit eingegangen. In Kapitel 2.3.1 werden die mit der Konzeption der Beschichtungssimulation in dieser Arbeit verfolgten Ziele spezifiziert und in Kapitel 2.3.2 verfügbare Simulatoren für thermische Spritzprozesse beschrieben.

2.1 Thermisches Spritzen

Thermische Spritzverfahren sind Verfahren zur Oberflächenbeschichtung von Bauteilen. Ihre Erfindung geht zurück auf den Schweizer Max Ulrich Schoop, welcher im Jahr 1909 das erste Patent zum thermischen Spritzen beim Berliner Patentamt einreichte [138] und im Jahr 1911 erstmals eine Korrosionsschutzschicht aus Zinn und Blei herstellte [11]. Heute sind thermische Spritzverfahren formal in der Norm DIN EN 657 [109] definiert als

„Verfahren, bei denen Spritzzusätze innerhalb oder außerhalb der Spritzpistole zugeführt und bis zum plastischen oder geschmolzenen Zustand aufgeheizt und dann auf die vorbereitete Oberfläche geschleudert werden; die Oberfläche wird dabei nicht aufgeschmolzen.“

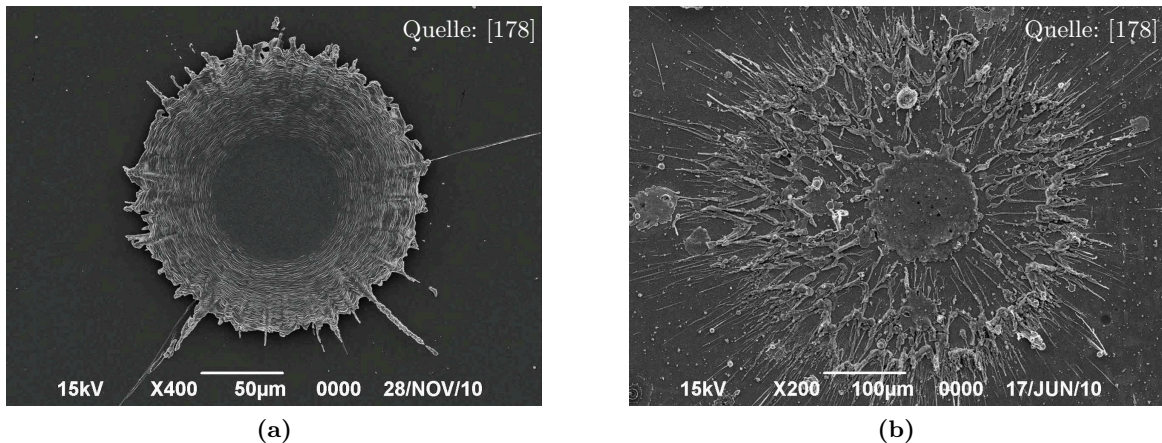


Abbildung 2.1: REM-Aufnahmen unterschiedlicher Splatformen¹. (a) Runder „Disc“- oder „Pancake“-Splat. (b) Zerspratzter, beim Aufschlag zerplatzter Splat.

Diese Definition umfasst ein weites Feld unterschiedlicher Spritzverfahren. Gemeinsam ist allen Verfahren die Aufheizung eines Spritzzusatzes oder auch Spritzwerkstoffes mit thermischer Energie in einer Spritzpistole oder einem Spritzbrenner und die Zuführung kinetischer Energie, um das (teil-)geschmolzene Material in Richtung der zu beschichtenden Oberfläche zu beschleunigen. Verfügbare Verfahren werden in der Regel nach der Art der Zuführung thermischer und kinetischer Energie sowie nach der Art der Zuführung des Spritzzusatzes klassifiziert. Dabei werden Spritzzusätze in Form von Pulvern oder Drähten, seltener als Schnüre oder Stäbe zugeführt. Die Aufheizung des Materials erfolgt in einer Flamme, einem Plasma, einem Lichtbogen oder durch elektrische Erwärmung [11]. Durch das Aufschmelzen des Spritzzusatzes oder die gezielte Zerstäubung mittels eines Zerstäubergases bilden sich feine Tröpfchenpartikel, welche durch die Verbrennungsgase des Prozesses oder ein separates Trägergas mitgerissen und beschleunigt werden. Dabei werden Flamm- oder Plasmaaustrittstemperaturen bis zu 8.300 °C [38] und Partikeltemperaturen zwischen 200 °C und 3.300 °C erreicht. Erzielbare Partikelgeschwindigkeiten beginnen mit den Flamm- oder Lichtbogenspritzverfahren bei 30–240 m/s und reichen bis zu mehrfacher Überschallgeschwindigkeit bei ca. 1.000 m/s für Kaltgas- oder Hochgeschwindigkeits-Flamm- oder Lichtbogenspritzverfahren (engl.: High Velocity Oxygen Fuel spraying, HVOF)[11, 38]. Tabellierte Aufzählungen verfügbarer Spritzverfahren und ihrer Eigenschaften finden sich beispielsweise in [38] und [11].

2.1.1 Partikelaufrall und Schichtaufbau

Beim Aufprall auf die Substratoberfläche wird die kinetische Energie der Partikel in Deformations- und Wärmeenergie umgewandelt und die Partikel breiten sich fladenförmig auf der

¹Die Abbildungen stellen Abwandlungen der Originalversionen von Kun Yang et al. [178] dar, verwendet entsprechend der Creative Commons Attribution License CC BY 3.0 (<https://creativecommons.org/licenses/by/3.0/>). Der Kontrast der Abbildungen wurde verbessert.

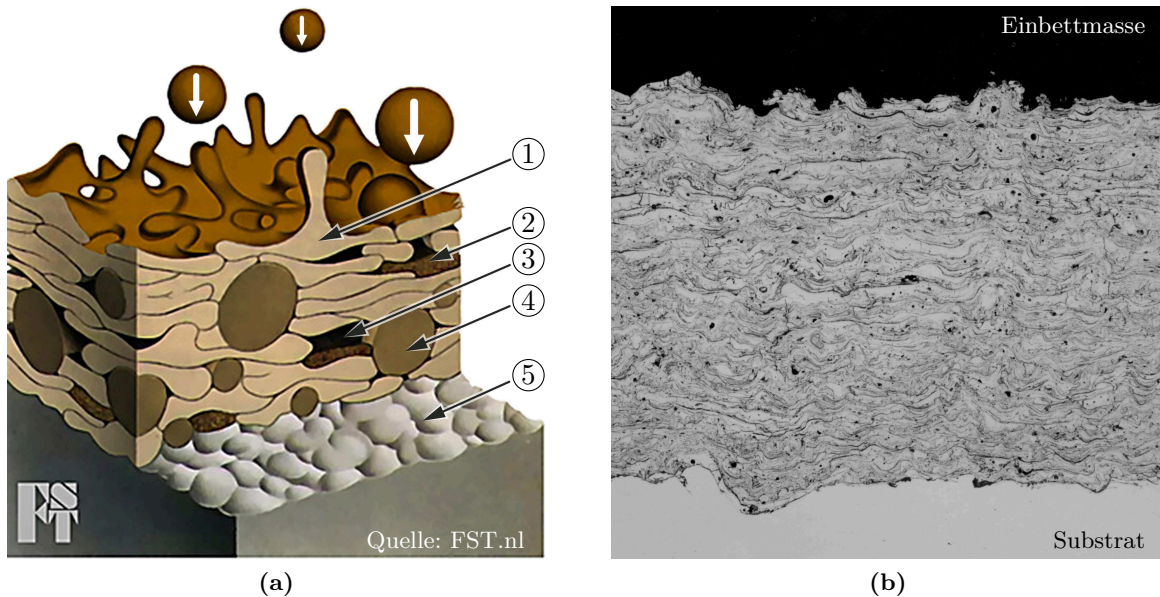


Abbildung 2.2: (a) Illustration des Schichtaufbaus (Quelle: FST.nl): ① Aufprallender Partikel. ② Oxideinschluss. ③ Pore. ④ Eingebetteter ungeschmolzener Partikel. ⑤ Aufgeraute Substratoberfläche. (b) Lichtmikroskopische Querschliffaufnahme einer Lichtbogen-gespritzten Beschichtung mit einer Schichtdicke von ca. 750 µm.

Oberfläche aus. Gleichzeitig kühlen die Partikel aufgrund der vergleichsweise geringen Substrattemperatur schlagartig ab und erstarren. Da hierbei sehr hohe Abkühlraten und damit verbundene Erstarrungszeiten im Bereich von 10^{-8} s bis 10^{-4} s erreicht werden, spricht man auch von „Rapid Solidification“ [11]. Nach dem Aufprall und der Erstarrung eines Partikels wird dieser in seiner ausgebreiteten Form als *Splat* bezeichnet. Der Verlauf des Ausbreitens und Abkühlens stellt einen komplexen Vorgang dar, welcher in Abhängigkeit von der Aufprallgeschwindigkeit, der Temperatur von Partikel und Substrat, der lokalen Topographie des Untergrundes sowie verschiedenen weiteren Parametern stark variiert. Als Resultat treten verschiedene Splatformen auf, die von kreisrunden, sogenannten „Pancake“- oder „Disc“-Splats, bis hin zu stark zerspratzten Formen (Abbildung 2.1) reichen, welche kaum noch als einzelner Splat erkennbar sind [114, 178]. Dieses Zersprätzen stellt eine extreme Form des sogenannten „impact splashing“ dar, welches einen Partikeleinschlag bezeichnet, bei dem sich kleine Tröpfchen ablösen und separat in der Umgebung abgelagert werden oder von der Oberfläche zurückprallen [56].

Durch das sukzessive gegenseitige Überlappen auftreffender Splats bildet sich auf dem Substrat eine porenbehaftete Beschichtungsmikrostruktur, in die auch un- oder teilgeschmolzene Partikel eingebettet sein können (Abbildung 2.2). Bei metallischen Spritzzusätzen kann es darüber hinaus vermehrt zur Ablagerung von Oxideinschlüssen kommen. Dies ist darin begründet, dass die im geschmolzenen Zustand reaktionsfreudigen Partikel, welche im fein zerstäubten Zustand ein hohes Verhältnis von Oberfläche zu Volumen aufweisen, an ihrer

Oberfläche sehr schnell oxidieren. Durch die gezielte händische oder robotergeführte Bewegung des Spritzbrenners über das Substrat werden typischerweise 50 µm bis 400 µm dicke Beschichtungen abgelagert [11], welche fest mit der Substratoberfläche verbunden sind. Die Haftung der Beschichtung auf der Substratoberfläche ist dabei in erster Linie auf die Verklammerung der einzelnen Splats untereinander und mit dem Substrat zurückzuführen. Der noch flüssige Schichtwerkstoff dringt in offene Poren, Risse und Hinterschnitte des Untergrundes ein und erstarrt dort formschlüssig. Eine Verstärkung der Schichthaftung auf dem Substrat wird ferner erreicht, wenn der Schichtwerkstoff Rauheitsspitzen des Substrats umfließt und beim Abkühlen aufgrund entstehender Schrumpfspannungen kraftschlüssig mit diesen verklammert [94]. Vor dem Beschichten eines Bauteils ist es daher in der Regel notwendig, glatte Oberflächen, beispielsweise durch einen Sandstrahlprozess, aufzurauen, um eine ausreichende Schichthaftung zu gewährleisten.

Zahlreiche Effekte, welche sich auf mikroskopischer Ebene beim Aufprall der Partikel und dem Aufbau der Schichtmikrostruktur abspielen, haben einen Einfluss auf makroskopische Eigenschaften des resultierenden Schichtverbunds. So bewirken beispielsweise die beschriebenen Splat-internen Zugspannungen auf mikroskopischer Ebene zwar eine gute Verklammerung der Beschichtung mit dem Substrat, makroskopisch können derartige, aufgrund unterschiedlicher thermischer Ausdehnungskoeffizienten von Schicht und Substrat entstehende Spannungen jedoch auch zu einer Verformung des Schicht-Substrat-Verbundes und im Extremfall zu einer Delamination der Schicht führen [183]. Ein weiteres Beispiel für den Einfluss mikroskopischer Effekte auf relevante makroskopische Parameter stellt der Grad des „impact splashing“ dar. Dieser beeinflusst nicht nur den Anteil des von der Oberfläche abprallenden Materials und damit den Haftwirkungsgrad und die Effizienz des Prozesses, sondern auch den Porositätsgrad der Beschichtung durch die Ausformung mehr oder weniger zerspratzter Splats, welche mehr respektive weniger Potential für Lufteinschlüsse aufweisen [2, 114, 178].

2.1.2 Industrielle Anwendungsgebiete

Nicht zuletzt aufgrund der hohen erreichbaren Prozesstemperaturen können in thermischen Spritzprozessen eine Vielzahl von Spritzzusatzwerkstoffen verarbeitet und auf eine ebenso große Anzahl unterschiedlicher Substratwerkstoffe aufgetragen werden. Zu den verarbeitbaren Schichtmaterialien gehören Metalle, Legierungen, intermetallische Verbindungen, Keramiken und Polymere (Kunststoffe) [38]. Darüber hinaus können Beschichtungen aus Verbundwerkstoffen sowie funktional gradierte Beschichtungen (engl.: Functionally Graded Materials, FGM) hergestellt werden. Übersichten über konkrete Werkstoffe und ihre Eigenschaften finden sich in [8, 11, 38, 115] sowie in der Norm DIN EN 1274 [110].

Entsprechend der großen Palette an Spritzzusätzen können für den industriellen Einsatz Beschichtungen mit einer Vielzahl unterschiedlicher Eigenschaften hergestellt werden, welche mechanischer, thermophysikalischer, elektrischer, magnetischer, optischer oder chemischer Natur sein können [115]. Im Rahmen des Beschichtungsprozesses werden diese Eigenschaften ganz oder teilweise auf die beschichteten Bauteile übertragen, um ihre funktionalen Eigenschaf-

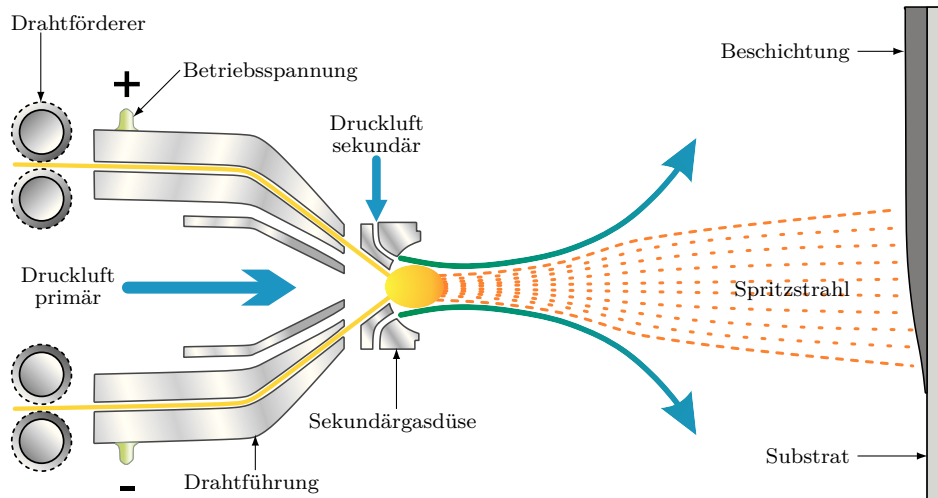


Abbildung 2.3: Schematische Darstellung einer Lichtbogenspritzpistole. Der Drahtförderer schiebt die Drähte durch die Drahtführungen, an deren Ende sich die unter Spannung stehenden Drahtenden treffen und in einem Lichtbogen abschmelzen. Geschmolzenes Material wird von der primärseitig zugeführten Druckluft mitgerissen und zerstäubt. Der entstehende Spritzstrahl wird mit der durch die Sekundärgasdüse zugeführten Druckluft eingeschnürt.

ten im Hinblick auf ihren Einsatzzweck zu erweitern oder zu verbessern [8]. Die wichtigsten industriellen Anwendungsgebiete sind nach [11]:

- Verschleißschutzschichten und reibmindernde Schichten für tribologische Systeme in allen Branchen,
- Korrosionsschutzschichten im Offshore-Bereich, bei Pumpen, Ventilen und Baukonstruktionen,
- Hochtemperaturkorrosionsschutzschichten und Wärmedämmschichten für Verbrennungskraftmaschinen (Turbinen- und Motorentechnik),
- Biokompatible Schichten für medizinische Implantate,
- Dichtungsbeläge in Flugtriebwerken.

Zum Einsatz kommen thermische Spritzprozesse hauptsächlich in den Branchen Luft- und Raumfahrttechnik, Automobilindustrie, Druck- und Papierindustrie, Fahrzeug- und Schiffbau, allgemeiner Maschinenbau, Kraftwerkstechnik, Bauindustrie und Medizintechnik.

2.1.3 Lichtbogenspritzen

Der Lichtbogenspritzprozess (engl.: Twin Wire Arc Spraying, TWAS) zeichnet sich dadurch aus, dass die in Form von Drähten zugeführten Spritzzusätze in der Spritzpistole mittels eines Lichtbogens, welcher unmittelbar zwischen den Drähten gezündet wird, abgeschmolzen werden. Abbildung 2.3 zeigt eine schematische Darstellung eines Lichtbogenspritzbrenners.

Die beiden Spritzdrahnte mit typischen Durchmessern zwischen 1,2 mm und 3,2 mm werden dabei mittels einer in der Regel aus Kupfer bestehenden Drahtfuhrung unter einem Winkel von 30° bis 60° aufeinander zu gefuhrt [11]. Durch Anlegen einer elektrischen Spannung an die Drahtfuhrung und somit an die Spritzdrahnte entsteht durch einen Kurzschluss am Beruhrpunkt der Drahtenden ein bis zu 6.100 K heier Lichtbogen [115], welcher die Drahtenden aufschmilzt. Das im Lichtbogen geschmolzene Material wird mit Hilfe eines Zerstaubergases mitgerissen, zerstaubt und in Richtung des Substrats beschleunigt. Dieses Zerstaubergas wird auch als Primargas bezeichnet. Zum Einsatz kommt hierfur zumeist Druckluft oder Stickstoff. Der kontinuierliche Vorschub des Spritzdrahtes durch einen Drahtforderer, welcher sich in der Spritzpistole, in einer externen Basiseinheit oder in beiden befindet, verhindert ein Abreien des Lichtbogens und ermoglicht einen kontinuierlichen Abschmelzvorgang. Die an den Drahnten anliegenden Spannungen fallen ublicherweise in den Bereich von 18–45 V, die Stromstarken in den Bereich von 200–350 A. Zur weitergehenden Beschleunigung, zur Einschnurung sowie gegebenenfalls zur atmospharischen Abschirmung des Spitzstrahls kann eine Sekundargasduse, wie in Abbildung 2.3 dargestellt, eingesetzt werden. Die Partikel erreichen dabei Geschwindigkeiten zwischen 100 m/s und 150 m/s bei einem typischen Spritzabstand von 50 mm bis 170 mm [11, 115].

Im Vergleich zu anderen Spritzprozessen zeichnet sich der Lichtbogenspritzprozess durch eine besonders gute Energieeffizienz aus, da das Spritzmaterial hier nicht in einer Flamme oder einem Plasma geschmolzen wird, bei denen sich nur ein geringer Teil der thermischen Energie auf die Spritzwerkstoffe ubertragt. Daruber hinaus tragen auch die Verwendung vergleichsweise kostengunstiger Prozessgase, welche hier nur die kinetische Energie liefern mussen, sowie die erzielbaren hohen Auftragsraten von bis zu 60 kg/h zur hohen Wirtschaftlichkeit des Verfahrens bei. Nachteilig ist hingegen die prinzipbedingte Beschrankung des Verfahrens auf elektrisch leitfahige Spritzzusatze. Eine geringfugige Flexibilisierung erlaubt an dieser Stelle jedoch die Verwendung von Fulldrahnten, welche aus einem metallischen Mantel und einer pulverformigen Fullung eines beliebigen Werkstoffs bestehen.

2.2 Maschinenbaulicher Anwendungshintergrund dieser Arbeit

Den maschinenbaulichen Anwendungshintergrund fur diese Arbeit stellt die Beschichtung von Werkzeugsystemen fur die Umformung von Stahlblechen dar, wie sie beispielsweise in der Automobilindustrie zur Fertigung von Karosserieteilen eingesetzt werden. Stark vereinfacht dargestellt werden bei diesen Verfahren initial flache Stahlbleche in einer Presse mittels eines beweglichen Stempels in eine Form, welche auch Matrize genannt wird, gezogen. Das Blech wird dabei auerhalb der Hohlform durch den sogenannten Niederhalter auf die Randbereiche der Matrize gedruckt, um ein kontrolliertes Gleiten des Blechs uber den Ziehradius in die Form zu ermoglichen und eine Faltenbildung zu vermeiden. Fur eine genauere Beschreibung des Verfahrens sei auf die Fachliteratur verwiesen [14, 88].

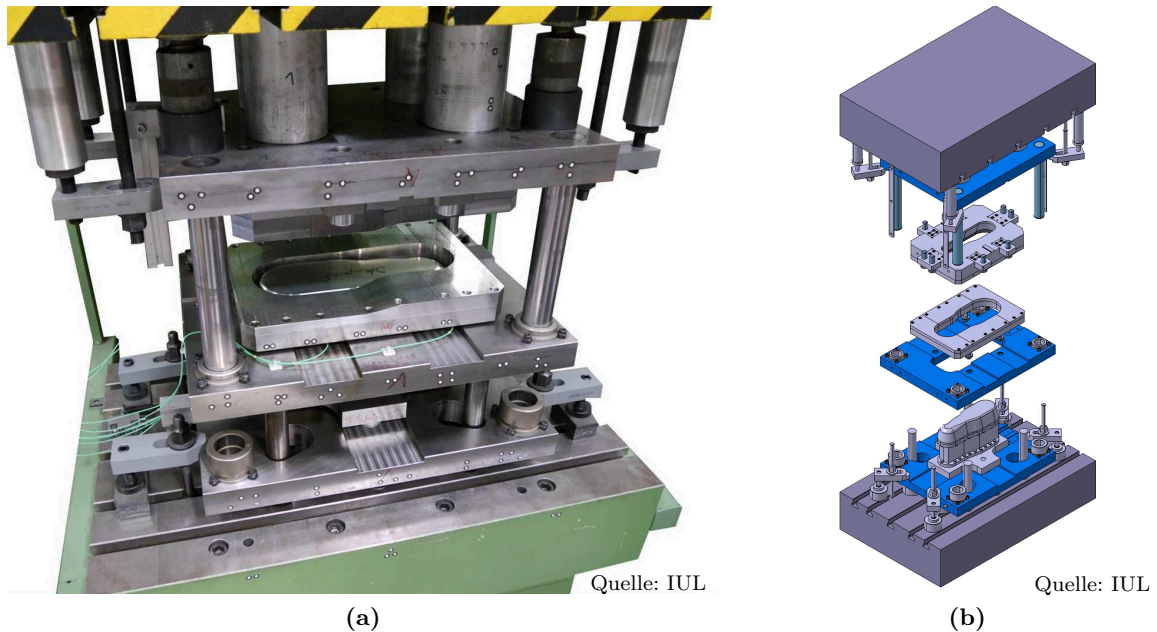


Abbildung 2.4: Im Rahmen des SFB 708 eingesetztes Tiefziehwerkzeug (Quelle: IUL, www.iul.eu). Matrize, Niederhalter und Stempel des Werkzeugs werden mit Verschleißschutzschichten verstärkt. Im Rahmen dieser Arbeit wurden zwei Segmente der Matrize zur Validierung der Simulation beschichtet (Kapitel 6.3). (a) Foto des Werkzeugaufbaus. (b) Schematische Explosionszeichnung des Werkzeugsystems.

Im Rahmen des Sonderforschungsbereichs 708 werden in diesem Zusammenhang Verfahren zur robotergestützten Beschichtung der komplexen Tiefziehwerkzeuge mit Verschleißschutzschichten erforscht. Übergeordnetes Ziel ist es dabei, den durch die großen Prozesskräfte auftretenden Werkzeugverschleiß an Matrize, Niederhalter und Stempel zu reduzieren und so die Haltbarkeit und Standzeit der teuren Werkzeuge zu erhöhen. Eine besondere Bedeutung kommt in diesem Beschichtungsprozess der Einhaltung enger Toleranzen für die Schichtdickenverteilung zu, da bereits minimale Abweichungen einen großen Einfluss auf die auftretenden Prozesskräfte und das Gleit- und Reibverhalten des Blechs haben [155]. Zur Korrektur von (positiven) Schichtdickenabweichungen sowie zur Einstellung der für den Umformprozess benötigten tribologischen Oberflächeneigenschaften kann zwar ein anschließender Schleifprozess eingesetzt werden [117], der Einsatz dieses Prozesses ist jedoch, insbesondere im Hinblick auf die Verschleißbeständigkeit der Beschichtung, sehr aufwendig. Eine derartige Nachbearbeitung sollte daher durch das möglichst exakte Erreichen der Zielschichtdicke vermieden oder zumindest minimiert werden.

Eingesetztes Beschichtungsverfahren

Zur Beschichtung der Tiefziehwerkzeuge wird im Rahmen des SFB 708 unter anderem ein Lichtbogenspritzprozess eingesetzt, welcher in dieser Dissertation als Basis für die Kalibrierung und Validierung der realisierten Beschichtungssimulation dient. Zum Einsatz kommt hier



Abbildung 2.5: Spritzpistole des SmartArc Systems montiert auf einem Roboter.

eine Lichtbogenspritzanlage vom Typ SmartArc™ der Firma Oerlikon Metco² (vormals Sulzer Metco). Abbildung 2.5 zeigt die auf einem Roboter montierte Spritzpistole SmartArc PPG, Aufnahmen vom laufenden Beschichtungsprozess sind in Kapitel 6 zu finden (Abbildung 6.9). Als Spritzzusatz werden Fülldrähte vom Typ DURMAT® AS-850 der Durum Verschleißschutz GmbH³ eingesetzt. Diese Drähte besteht aus einer Fe-Basis-Legierung als Hülle und sind mit Wolframschmelzkarbiden (WSC) mit einer Größe von 45–106 µm gefüllt (Abbildung 2.6). Diese extrem harten Karbide werden durch den Spritzprozess in einer duktilen Eisen-basierten Schichtmatrix eingebettet und tragen somit signifikant zu einer Erhöhung der abrasiven Verschleißfestigkeit der Beschichtung bei. Die Eigenschaften und die genaue Zusammensetzung dieser Drähte sind in Tabelle 2.1 angegeben.

Eigenschaften		Zusammensetzung (Gew.-%)					
Bezeichnung	Durmat AS-850	WSC	Cr	C	Si	Mn	Fe
Drahtdurchmesser (mm)	1,6	50	4	2	1,4	<1	Rest
Härte (HRC)	58-60						

Tabelle 2.1: Spritzwerkstoff Durmat AS-850³.

Im Rahmen einer umfassenden Parameterstudie [85] wurden die in Tabelle 2.2 angegebenen Spritzparameter für den Prozess ermittelt, welche die Standardkonfiguration für alle im Rahmen dieser Arbeit angefertigten Beschichtungen darstellt. Zur Fokussierung des Spritzstrahls wird eine Sekundärgasdüse eingesetzt, welche, wie der Primärgasstrom, mit Druckluft betrieben wird. Insbesondere für die Bahnplanung, aber auch für die in Kapitel 3 beschriebenen

²<http://www.oerlikon.com>

³<http://www.durmat.com>

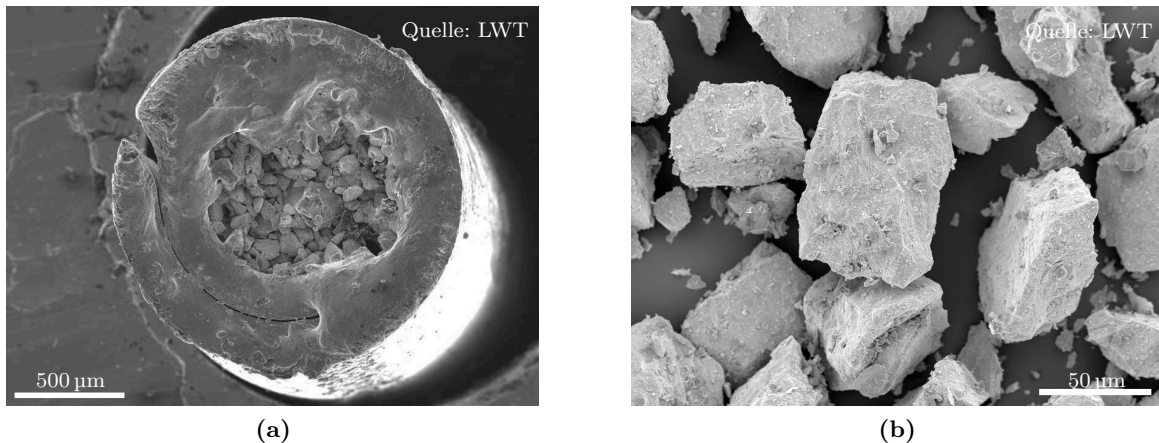


Abbildung 2.6: Fülldraht Durmat AS-850 (Quelle: LWT). (a) Rasterelektronenmikroskopaufnahme (REM) des verwendeten Fülldrahts im Querschnitt. Die Hülle besteht aus einer FE-Basis-Legierung und die Füllung aus Wolframschmelzkarbiden in Pulverform. (b) REM-Aufnahme der Karbidfüllung.

Footprintversuche ist der angestrebte optimale Spritzabstand von 110 mm ein wichtiger Parameter.

Zur Präparation der Proben wurden alle zu beschichtenden Substratoberflächen vor dem Spritzen mit Edelmetall gestrahlt, um die Oberfläche aufzurauen und eine bessere mechanische Mikroverklammerung zwischen Substrat und Beschichtung und damit insgesamt eine bessere Schichthaftung zu erreichen. Anschließend wurden die Proben in einem ca. 10-minütigen Ethanol-Ultraschallbad gereinigt und in einem Ofen auf ca. 70–100 °C für die Beschichtung vortemperierte. Tabelle 2.2 fasst die verwendeten Strahlparameter zusammen.

Spritzparameter		Strahlparameter	
Spritzabstand (mm)	110	Strahlgut	Edel-Korund EKF-14
Lichtbogenstrom (A)	220	Körnung (µm)	1180-1700
Lichtbogenspannung (V)	30	Druck (bar)	4,0
Primärgasdruck (bar)	6,0	Strahlabstand (mm)	100
Sekundärgasdruck (bar)	4,0	Strahlwinkel (°)	45
Drahtklemmung (bar)	1,2		

Tabelle 2.2: Standardkonfiguration für Lichtbogenspritzversuche [85].

2.3 Beschichtungssimulationen

Die computergestützte Simulation technischer Fertigungsprozesse verfolgt in der Regel das Ziel, mindestens eine für die Eigenschaften des zu fertigenden Produktes oder den Prozess selbst relevante Größe zu berechnen. Aus wirtschaftlicher Perspektive ergibt sich dadurch in vielen Fällen eine erhebliche Kostenersparnis, da teure Versuche zur experimentellen Bestim-

mung einer geeigneten Prozessparametrisierung vermieden oder zumindest in ihrer Anzahl oder im Umfang reduziert werden können. Aus wissenschaftlicher Perspektive bewirkt die Entwicklung einer Prozesssimulation eine Vertiefung des Verständnisses der simulierten Vorgänge. Neue Erkenntnisse können dabei sowohl direkt durch die für die Implementierung der Simulation notwendige Analyse des zu simulierenden Systems oder indirekt durch die Ausführung von Simulationen mit neuen Parametrisierungen gewonnen werden.

Die Anforderungen an eine Beschichtung sind so vielfältig wie die verschiedenen Anwendungsbereiche und können mechanischer, elektrischer, chemischer, thermophysikalischer, magnetischer oder sogar optischer Art sein [115]. Dementsprechend kann die Qualität einer thermischen Beschichtung im Allgemeinen nicht anhand einer einzelnen Größe beurteilt werden, sondern hängt von einer Vielzahl an Faktoren ab. Welche Qualitätsfaktoren letztlich zur Beurteilung der Gesamtgüte einer Beschichtung geeignet sind, kann nur vor dem Hintergrund des Zwecks der Beschichtung beziehungsweise des Anwendungsfalls beurteilt werden. So fördert beispielsweise eine niedrige Porosität die Schichtkohäsion und -härte, was für die Herstellung von Verschleißschutzbeschichtungen und für schleifend nachzubearbeitende Beschichtungen vorteilhaft ist. Dem entgegen ist für Wärmedämmschichten oder für Beschichtungen von Festschmierstofflagern eine höhere Porosität erstrebenswert, um die Isolationsfähigkeit oder die Aufnahmefähigkeit von Schmierstoffen zu steigern [38]. Trotz der Vielfalt der an die Beschichtungen gestellten Anforderungen basieren viele Qualitätsfaktoren jedoch auf einer beschränkten Anzahl an unterschiedlich gewichteten Grundgrößen, wie beispielsweise der Schichtdicke, dem Spritzwinkel oder dem Temperaturverlauf während des Prozesses.

Im Rahmen der Beschichtung von Tiefziehwerkzeugen sind, wie bereits in Kapitel 2.2 diskutiert wurde, eine gleichmäßige Schichtdickenverteilung, eine hohe Härte sowie eine hohe Verschleißfestigkeit der Beschichtung von Bedeutung. Maßgeblich für eine hohe Härte und Verschleißfestigkeit ist dabei ein dichtes, porositätsarmes Mikrostrukturgefüge, welches unter anderem durch die Vermeidung flacher Spritzwinkel begünstigt wird [154]. Auch in anderen Anwendungsgebieten wie zum Beispiel der Applikation von Wärmedämmschichten auf Turbinenschaufeln [121], von Verschleißschutz- und Dichtungsschichten in Pumpen [23] oder der Beschichtung von Rollen und Walzen für die Papierindustrie [75] treten enge Toleranzforderungen für die Schichtdicken auf.

2.3.1 Ziele der konzipierten Simulation

Das primäre Ziel der im Rahmen dieser Dissertation konzipierten Simulation ist daher die Berechnung der Schichtdickenverteilung auf dreidimensionalen Bauteiloberflächen in Abhängigkeit einer vorgegebenen Roboterbahn. Dabei wird eine möglichst breite Anwendbarkeit der Simulation durch folgende Teilziele angestrebt:

- **Bauteilgeometrie**

Als Bauteilgeometrien werden beliebig komplexe dreidimensionale Freiformgeometrien, wie sie auch bei der Konstruktion von Umformwerkzeugen Verwendung finden, unterstützt.

- **Roboterbahn**

Die vorzugebenden Roboterbahnen werden in keiner Weise beschränkt und beliebige Bahnverläufe zugelassen.

- **Beschichtungsmodell**

Es werden keine einschränkenden Annahmen für die abzubildende Spritzcharakteristik getroffen. Insbesondere erfolgt keine Beschränkung auf gaußförmige oder rotationssymmetrische Charakteristiken.

Zur Sicherstellung der Qualität abgelagerter Schichtdicken existieren für viele Spritzprozesse spezifische Vorgaben zu einzuhaltenden Eingriffsparametern. Um eine dementsprechende Beurteilung der Qualität einer Roboterbahn zu ermöglichen, werden zusätzlich zur Schichtdicke sekundäre Zielgrößen berechnet. Im Einzelnen werden folgende Größen berechnet:

1. Schichtdickenverteilung auf der Bauteiloberfläche (primäre Zielgröße)
2. Sekundäre, oberflächenbezogene Zielgrößen:
 - Durchschnittlicher gewichteter sowie maximaler Spritzwinkel
 - Durchschnittlicher gewichteter sowie minimaler und maximaler Spritzabstand
 - Umgebungstemperatur
3. Bahnbezogene Zielgrößen:
 - Prozesseffizienz
 - Volumenablagerungsrate
 - Durchschnittlicher gewichteter sowie maximaler Spritzwinkel
 - Durchschnittlicher gewichteter sowie minimaler und maximaler Spritzabstand

Dabei wird zwischen oberflächen- und bahnbezogenen Zielgrößen unterschieden. Oberflächenbezogene Größen werden an diskreten Stellen auf der Oberfläche des gegebenen Bauteils berechnet und dienen der Identifikation lokaler Abweichungen von der Zielschichtdicke beziehungsweise von optimalen Eingriffsparametern. Die Mittelwertbildung sowie die Bestimmung von Minima und Maxima erfolgt für die oberflächenbezogenen Größen über die diskreten Simulationszeitpunkte und somit die Positionen des Spritzbrenners entlang der Roboterbahn. Die berechneten oberflächenbezogenen Umgebungstemperaturen stellen die durch den über die Oberfläche bewegten Spritzbrenner an verschiedenen Stellen anliegenden Temperaturen dar. Diese können als Randbedingungen für eine externe Simulation der Temperaturentwicklung im Bauteil verwendet werden. Eine solche thermodynamische Simulation wurde im Rahmen einer Kooperation erstellt und in [10] dokumentiert, ist jedoch nicht Gegenstand dieser Arbeit.

Bahnbezogene Größen werden für diskrete Positionen des Spritzbrenners entlang der Roboterbahn berechnet und geben so die zeitabhängigen Verläufe der Prozesseffizienz und der

genannten Eingriffsbedingungen wieder. Die Bildung der Durchschnitts-, Minimal- und Maximalwerte erfolgt für diese Größen über alle diskreten Punkte der Bauteiloberfläche, welche sich zu dem jeweiligen Zeitpunkt im Eingriff befinden. Bahnbezogene Zielgrößen erlauben eine direkte Bewertung der Qualität einzelner Positionen oder Abschnitte der verwendeten Roboterbahn und sind daher besonders für den Einsatz der Simulation zur Bahnplanung wichtig, welcher ein weiteres Hauptziel der Simulation darstellt. Der Grundgedanke besteht dabei darin, die Roboterbahn für die Beschichtung eines Bauteils in einem iterativen Verfahren durch sukzessives Simulieren der Beschichtung und Anpassen der Bahn zu verbessern. Die bahnbezogenen Größen liefern in diesem Kontext einen Indikator, welche Abschnitte der Bahn optimierungsbedürftig sind, also zum Beispiel zu flache Einschlagwinkel oder eine stark abweichende Spritzdistanz aufweisen.

Durch den Einsatz der Simulation zur automatisierten Optimierung von Roboterbahnen ergeben sich besondere Anforderungen an die Laufzeit der Simulation. Dies ist darauf zurückzuführen, dass die Beschichtungssimulation in jeder Iteration der Bahnplanung ein oder mehrmals ausgeführt werden muss, um die Auswirkungen von Bahnmodifikationen auf die Zielgrößen zu bestimmen oder um Ableitungen nach Optimierungsparametern durch numerische Differentiation zu berechnen. Um eine akzeptable Laufzeit des gesamten Optimierungsverfahrens zu gewährleisten, wird daher einer effizienten Implementierung der Beschichtungssimulation explizit ein hoher Stellenwert beigemessen. Um dieses Ziel zu erreichen, wird eine möglichst konsequente Nutzung GPU-basierter Algorithmen angestrebt.

2.3.2 Verfügbare Simulationssoftware

Simulationssoftware zur Berechnung der Schichtdickenverteilung für thermische Spritzprozesse findet sich im kommerziellen Bereich in Verbindung mit Software zur Roboterbahnplanung. Im akademischen Bereich existieren Publikationen zu Entwicklungen kleinerer, dedizierter und in der Regel nicht veröffentlichter Softwaretools zur Schichtdickenberechnung verschiedener Forschergruppen.

Kommerzielle Simulationssoftware

Zentrales kommerzielles Anwendungsgebiet für Beschichtungssimulationen thermischer Spritzprozesse, welche die Schichtdickenverteilung auf komplexen Bauteilen berechnen, stellt die Prozessplanung beziehungsweise die Off-Line-Programmierung (OLP) der Beschichtungsroboter dar [121]. In diesem Bereich existieren mehrere Softwaretools, welche Methoden zur mehr oder weniger automatisierten Erstellung von Roboterbahnen zumeist auf Basis von CAD-basierten Bauteilgeometrien bereitstellen. Zu diesen zählen herstellerspezifische Softwaresysteme, wie beispielsweise RobotStudio⁴ von ABB oder RoboGUIDE⁵ von Fanuc, sowie

⁴ABB Ltd. (Asea Brown Boverly), <http://new.abb.com/products/robotics/robotstudio>

⁵FANUC Corporation, <http://robot.fanucamerica.com/products/vision-software/>

Drittanbietersoftware wie Robcad⁶ (Siemens), Delmia/Robotics⁷ (Dassault Systèmes), RobotWorks⁸ (Compucraft Ltd.), Eureka⁹ (Roboris) oder Robotmaster¹⁰ (Jabez Technologies). Kernkompetenz bei diesen Produkten ist die Erstellung von Roboterbahnen sowie deren Simulation im Hinblick auf Kollisionsfreiheit und dynamische Effekte. Die großen Anbieter, Siemens und Dassault Systèmes, integrieren ihre OLP-Software darüber hinaus in äußerst komplexe PLM-Systeme (engl.: Product Lifecycle Management), welche alle Schritte des Lebenszyklus eines Produktes von Entwurf und Konstruktion bis hin zur Produktionsplanung und Produktwartung verwaltungstechnisch unterstützen. Im Kontext dieser umfassenden Lösungen, Tecnomatix von Siemens PLM und DELMIA (engl.: Digital Enterprise Lean Manufacturing Interactive Application) von Dassault Systèmes, werden auch Methoden zur Simulation verschiedenster Prozesse wie Punkt- und Bahnschweißen, Bohren, Nieten, Schneiden, Polieren oder Lackieren angeboten. Im Vergleich zu den auch bei kleineren Softwareanbietern weit verbreiteten Unterstützungen für Punkt- oder Bahnschweißen stellt die Simulation thermischer Spritzprozesse jedoch keinen klassischen Anwendungsfall dar. Ein verwandtes und weit verbreitetes Verfahren ist hingegen das robotergestützte Spritzlackieren, welches von dem Modul Robcad Paint, von DELMIA über die Drittanbietersoftware Fastsurf (Cenit AG) sowie von der Software Eureka unterstützt wird. In diesem Kontext bewerben sowohl Siemens als auch Cenit die Möglichkeit, durch eine entsprechende Konfiguration der Lackiersimulation auch thermische Spritzprozesse simulieren zu können. Detaillierte Informationen zur Funktionsweise der in diesen Produkten integrierten Simulationen werden nicht veröffentlicht und sind auch auf Anfrage nicht erhältlich, so dass für die Produkte Eureka und Fastsurf an dieser Stelle auf die spärlichen Informationen aus Produktbroschüren zurückgegriffen werden muss. Für Robcad Paint gilt dies prinzipiell auch, jedoch wurde kürzlich die Dissertation [121] von A. Sadovoy zu dem Thema „Modeling and offline simulation of thermal spray coating process for gas turbine applications“ veröffentlicht. A. Sadovoy arbeitet als Beschichtungsexperte bei Siemens und bietet in seiner Dissertation Einblicke in Simulationsmethoden, welche zumindest prototypisch in Robcad Paint implementiert wurden. Ob diese Methoden tatsächlich in Zukunft von Siemens verwendet werden, ist nicht sicher, kann aber als wahrscheinlich gelten. Im Folgenden werden die bekannten Eigenschaften der drei Softwaresysteme beschrieben:

Eureka. Kernkompetenz der Software Eureka ist die Erstellung und Optimierung von Roboterbahnen auf Basis von CAM-generierten Bahnen für CNC-Maschinen. In diesem Kontext kann in virtuellen Roboterzellen unter anderem auch der Materialabtrag in Fräsprozessen simuliert werden. Für Lackierprozesse bietet die Software eine Visualisierung des Materialauftrags auf der Werkstückoberfläche, wobei als Simulationsbasis Spritzmuster dienen, welche mit mehreren Parametern – welche genau bleibt unklar – definiert werden können. Die Berechnungen der Schichtdickenverteilung berücksichtigen zumindest die essentiellen Parameter

⁶Siemens PLM, <http://www.plm.automation.siemens.com>

⁷Dassault Systèmes, <http://www.3ds.com/de/produkte-und-services/delmia/>

⁸Compucraft Ltd., <http://www.robotworks-eu.com/>

⁹Roboris, <http://www.roboris.it/en/product/eureka-robot>

¹⁰Jabez Technologies, <http://www.robotmaster.com/>

Geschwindigkeit, Abstand und Sprühwinkel. Eine über die Visualisierung der Schichtdicken hinausgehende Verwendung der Simulation, beispielsweise zur automatischen Optimierung von Roboterbahnen, ist nicht erkennbar.

Fastsurf. Fastsurf ist eine in DELMIA integrierte Lösung zur Offline-Programmierung von Robotern für Sprühapplikationen. Primär wird Fastsurf mit einem Prozessmodul für Lackieren vertrieben, Weiterentwicklungen für weitere Sprüh- oder Spritzanwendungen sollen auf Anfrage möglich sein. Für das Lackiermodul lassen sich kreisförmige und elliptische Pinselprofile definieren, wobei auch mehrere Profile pro Werkzeug möglich sind. Verschiedene randkurven- oder flächenbasierte Bahngenerierungsmethoden werden angeboten. Zwar wird in der Produktbeschreibung von Bahnplanung und -optimierung gesprochen, aber auch bei dieser Software bleibt offen, ob die Simulation zur automatisierten Optimierung der Bahnen verwendet wird. Genutzt wird dieses Tool unter anderem von ALSTOM zur Simulation des Schichtauftrags von APS- und LPPS-Prozessen (engl.: Atmospheric Plasma Spray, APS; Low Pressure Plasma Spray, LPPS), wobei als Pinselformen ausladende mehrdimensionale Lookup-Tabellen zum Einsatz kommen, welche die Schichtwachstumsrate in Abhängigkeit des Spritzabstands, des Spritzwinkels und der zweidimensionalen Position in einer Substratebene beschreiben [128]. Mit Hilfe eines externen Tools werden dabei Pinselbibliotheken mit Spritzmustern anhand von eingestellten Prozessparametern generiert.

Robcad Paint. Das Modul Robcad Paint erweitert Robcad um Funktionalitäten zur Realisierung von Lackierprozessen. Diese umfassen unter anderem spezielle Funktionen zur Bestimmung der Roboterbahn und eine Funktion zur Simulation der Schichtdicken für Lackiermaschinen unter Verwendung konventioneller (druckluftbetriebener) oder elektrostatischer Lackierpistolen. Auch für Robcad Paint wird eine einfache Anpassung der Simulationsmethoden für weitere Prozesse, darunter auch thermische Spritzprozesse, beworben.

Im Rahmen der Dissertation [121] wurde zeitgleich und unabhängig von den Arbeiten in der vorliegenden Dissertation eine Verfahrensweise zur Simulation thermischer Spritzprozesse mit Robcad Paint entwickelt. Das von A. Sadovoy entwickelte grundlegende Beschichtungsmodell basiert dabei auf einer komplexen analytischen Formulierung für bivariate gaußförmige Spritzprofile. Diese sind über ihre maximale Höhe, achsenparallele Standardabweichungen sowie einen zweidimensionalen Offset und einen Rotationswinkel in der zur Spritzachse orthogonalen Ebene definiert. Des Weiteren wird eine Formulierung für den winkel- und distanzabhängigen Haftwirkungsgrad einbezogen. Basierend auf dieser dreidimensionalen Formulierung des Spritzprofils wird zunächst eine Formulierung für das zweidimensionale Querschnittsprofil orthogonal zu einer geradlinigen Verfahrensrichtung der Spritzpistole hergeleitet. Anhand additiver Überlagerungen dieses zweidimensionalen Profils werden analytische Lösungen für die Schichtdicke und Welligkeit des Profils bei Mäanderbahnen mit variablem Bahnabstand auf ebenen und zylindrischen Geometrien ermittelt und experimentell validiert. Die Implementierung des dreidimensionalen Modells in Robcad Paint wird im Rahmen dieser Arbeit nicht ausführlich beschrieben und demonstrierte Simulationsfälle beschränken sich bis auf eine einzige Ausnahme auf ebene Geometrien und sehr einfache Bahnen. Für das Beispiel einer freigeform-

ten Turbinenschaufel werden Abweichungen zwischen Simulation und Messung im Bereich bis 20 % (durchschnittlich etwa 7 %) für 15 nicht näher spezifizierte Messpunkte angegeben. GPU-beschleunigte Simulationsverfahren kommen hier nach Kenntnis des Autors nicht zum Einsatz.

Alle genannten Softwaresysteme befassen sich nicht nativ mit der Simulation thermischer Spritzprozesse, sondern primär mit Lackieranwendungen, wobei neben generellen Bahnplanungsmethoden die Optimierung von Fertigungszyklen und Taktzeiten im Vordergrund steht. Das Werben der Hersteller für die Anpassbarkeit der (nicht öffentlichen) Prozessmodelle auf viele weitere Oberflächenprozesse scheint vor dem Hintergrund der vorhandenen umfangreichen PLM-Softwareumgebungen zumindest teilweise gerechtfertigt, da diese sicherlich eine gute Softwarebasis für die Implementierung weiterer Prozesse bieten. Andererseits ist die Marketingfunktion derartiger Aussagen nicht zu unterschätzen und es ist aus wissenschaftlich technologischer Sicht nicht davon auszugehen, dass eine Anpassung der Modelle durch eine einfache Neuparametrisierung geleistet werden kann, wie die Dissertation von A. Sadovoy zeigt [121]. In [128] wird der Entwicklungsstand verfügbarer Simulationssysteme darüber hinaus als „eher stark beschränkt und experimentell als vollwertig kommerziell“ eingestuft.

Im Gegensatz zu diesen Systemen ist die im Rahmen dieser Arbeit entwickelte Simulation explizit auf eine Unterstützung der Fertigungs- beziehungsweise Bahnplanung für thermische Spritzprozesse ausgerichtet und umfassend auf eine hohe Simulationsgeschwindigkeit optimiert. Darüber hinaus bietet das konzipierte Footprint-basierte Beschichtungsmodell eine freiere Definition der Prozesscharakteristik. Auch gegenüber der Dissertation [121] wird mit diesem Modell ein anderer Berechnungsansatz gewählt und umfassender evaluiert. Ferner werden in Kapitel 3 dieser Arbeit bereits Methoden für eine robustere Messung von Footprintprofilen und die damit verbundene Kalibrierung der Simulation präsentiert, welche in [121] als relevantes zukünftiges Forschungsgebiet für eine Steigerung der Zuverlässigkeit der Beschichtungssimulation genannt werden.

Simulationsmethoden in der wissenschaftlichen Literatur

Im Gegensatz zu kommerziellen Systemen befassen sich wissenschaftliche Publikationen im Themenbereich der Simulation von Schichtdickenverteilungen für thermische Spritzprozesse mit einzelnen Aspekten der Modellierung oder der Modellierung für einen bestimmten Prozess oder Prozesstyp. Die publizierten Methoden zur Simulation der Schichtdicken ähneln sich dabei insofern, dass typischerweise zunächst eine mathematische Modellierung des Massenflusses- oder des Schichtwachstums vorgenommen wird, welche in einem zweiten Schritt zur Berechnung von Schichtdicken auf Bauteilgeometrien eingesetzt wird. Die angewendeten Verfahrensweisen und Detailzielsetzungen einzelner Forschergruppen unterscheiden sich jedoch deutlich voneinander.

Trifa et al. entwickeln ein modellbasiertes Expertensystem zur Vorhersage der Form zweidimensionaler Spritzprofile als Querschnittsprofile von Linearbahnen [157, 158]. Diese Vorhersage basiert auf einer Datenbank von experimentell ermittelten Zusammenhängen zwischen

Eingriffsparametern wie Spritzwinkel, Spritzabstand, Verfahrensgeschwindigkeit und Bahnoffset einerseits sowie verschiedener Parameter einer gaußförmig angenommenen Profilform wie Maximalhöhe, Querschnittsfläche, Streuung, Offset und Schiefe andererseits [159]. Demonstriert wird die Schichtdickenberechnung auf Basis der vorhergesagten Profile für planparallel zu einer ebenen Substratgeometrie verlaufenden Mäanderbahn mit variablem Bahnabstand. Die Berechnung erfolgt durch Addition der entsprechend des Bahnabstands gegeneinander verschobenen Querschnittsprofile. Deng et al. entwickeln das Robotstudio-Plugin „Thermal Spray Toolkit“ [40, 42, 53], welches die Berechnung von Schichtdicken ebenfalls für zweidimensionale Querschnitte von Mäanderbahnen realisiert. Einstellbare Parameter sind hier der Bahnabstand der Mäanderbahn (engl.: scannig step) und der Spritzabstand. Die überlagerten Profile werden als zuvor auf experimentelle Daten gefittete Gaußfunktionen aus einer Datenbank geladen. Einen ähnlichen Grundansatz verfolgen auch Fasching et al. [54], allerdings werden hier keine symmetrischen Einzelbahnprofile vorausgesetzt. Stattdessen wird zunächst ein optimaler Anstellwinkel der Spritzpistole berechnet, unter dem gemessene Profile, welche initial zur Spritzachse des Brenners unsymmetrisch sind, auf einer ebenen Substratoberfläche möglichst symmetrisch abgelagert werden. Die Simulation der Beschichtung erfolgt anschließend wiederum durch die Addition dieser 2D-Profile, wobei Mäanderbahnen mit unterschiedlichen Offsets zwischen einzelnen Überläufen zur Reduzierung der Welligkeit der Beschichtung untersucht werden. Ein einzelner Überlauf bezeichnet dabei die Ausführung einer vollständigen Mäanderbahn, welche die gesamte zu beschichtende Fläche abdeckt. Duncan et al. betrachten das Problem einen optimalen Bahnabstand für Mäanderbahnen zu ermitteln für ebene Substrate sowie rechtwinklige oder Überlagerungen von angewinkelten Mäanderbahnen im Frequenzbereich [48].

Einen anderen Ansatz zur Simulation verfolgen Djurić et al. mit der Publikationsreihe [44–47]. Darin wird ein zunächst zweidimensionales, später komplexes dreidimensionales Modell für den Massenfluss aufgestellt, welches mittels rekursiver Formulierungen nicht nur den Haftwirkungsgrad sondern auch die lokale Neuablagerung des nicht haftenden Materials in Form von Mikroquellen beschreibt. Notwendig wird diese komplexe Formulierung insbesondere vor dem Hintergrund, das Schließen kleiner Kavitäten in der Substratoberfläche und die Erzeugung von Makroporen zu modellieren. Die wachsende Oberfläche wird dabei durch eine Level-Set-Formulierung verfolgt. Nach [44] wird die Simulationsgeschwindigkeit insbesondere durch die aufwendige Sichtbarkeitsberechnung ausgebremst. Unklar ist bei diesem Ansatz ferner, wie die in dem Modell notwendigen Terme zur Neuablagerung von Material in der Praxis bestimmbar sind.

Die Anwendung einer dreidimensionalen Schichtauftragsberechnung findet sich weiterhin bei Lugscheider et al. [95, 96]. Ziel ist hier die Finite-Elemente-Simulation der Temperaturen und Spannungen in einem dreidimensionalen Bauteil-Schichtverbund. In diesem Ansatz wird zunächst eine dreidimensionale Wahrscheinlichkeitsverteilung für die Partikelflugrichtungen simulativ berechnet. Diese wird anschließend verwendet, um einzelne finite Elemente der im Vorhinein vernetzten Beschichtung entsprechend des Verfahrensweges des Roboters zu aktivieren und dadurch den Materialauftrag in die FE-Simulation einfließen zu lassen.

Bolot et al. [15] simulierten den Schichtauftrag auf einer ebenen Platte durch Superposition dreidimensionaler gaußförmiger Profile mit dem Fernziel, den somit berechneten Schichtauftrag in eine Temperatursimulation einzubeziehen.

Ferner gibt es einige Ansätze verschiedener Forschergruppen, in denen die Schichtdickenablagung und die Eingriffsbedingungen auf komplexen Bauteilen indirekt durch eine Optimierung der Roboterkinematik und -dynamik homogenisiert werden sollen [20, 41, 43, 59]. Eine Simulation der Schichtdicken findet hier nicht statt.

Betrachtet man die Eignung der dargestellten Arbeiten im Hinblick auf die in Kapitel 2.3.1 beschriebene Zielsetzung der in dieser Dissertation konzipierten Simulation, so wird deutlich, dass die in der Literatur beschriebenen Methoden wesentlichen Einschränkungen unterliegen. So verwendet die Mehrzahl der genannten Methoden lediglich zweidimensionale Modellierungen auf Basis überlagerter Einzelbahnprofile und fast alle Arbeiten modellieren den Massenfluss als rotationssymmetrisches Profil, welches zumeist als normalverteilt angenommen wird. Des Weiteren werden in vielen Publikationen stark vereinfachende Annahmen zur Form der Bauteile oder der Roboterbahnverläufe getroffen. Dabei werden vielfach nur ebene Bauteile sowie dazu planparallel verlaufende Mäander- oder Einzelbahnen betrachtet. Diese vereinfachenden Annahmen werden auch in die verwendeten Beschichtungsmodelle eingebracht, so dass beispielsweise die Modellierung von Bolot et al. [15] für flache Geometrien und planparallele Mäanderbahnen mit senkrechter Brenneranstellung und konstantem Abstand zur Substratoberfläche weder winkel- oder distanzabhängige Haftwirkungsgrade noch eine Abhängigkeit von der Spritzdistanz beinhaltet. Derartig vereinfachte Modelle sind für allgemeine, dreidimensionale Bauteile und Roboterbahnen nicht sinnvoll anpassbar.

Veröffentlichungen, in denen das Schichtwachstum für dreidimensionale Bauteile modelliert wird, finden sich auch im technologisch mit dem thermischen Spritzen verwandten Bereich des Spühkompaktierens (engl.: spray forming). In diesem Fertigungsverfahren wird eine flüssige Metallschmelze durch eine Düse versprüht und trifft auf eine Auffangfläche, wo die Schmelztröpfchen sukzessive zu einem festen Körper erstarren. Durch eine langsame Rotation und das kontinuierliche Wegziehen der Auffangfläche von der Düse wächst so ein näherungsweise zylindrischer Körper heran [65]. Durch die im Vergleich zu thermischen Spritzverfahren extrem hohen Verdüsungsraten von bis zu 2 kg/s [19] können so metergroße Rohlinge entstehen. Entsprechend dieses extremen Schichtwachstums befassen sich Publikationen zur Simulation des Schichtauftrags für das Sprühkompaktieren in der Regel mit dreidimensionalen Massenflussformulierungen und dreidimensional modelliertem Schichtwachstum [33, 34, 69, 92, 98, 132, 177]. Diese Publikationen nutzen alle ein rotationssymmetrisches gaußförmiges Beschichtungsmodell, welches vermutlich auf [132] zurückgeht. Aufgrund des Wachstums der erstellten Rohlinge sind Formulierungen für die Sichtbarkeitsberechnung Bestandteil der Simulationen. Erstaunlicherweise nutzt lediglich die älteste der genannten Veröffentlichungen von Seok et al. [132] ein GPU-beschleunigtes Tiefenpufferverfahren zur Sichtbarkeitsberechnung. Die übrigen Arbeiten wenden in der Regel einfache Backface-Culling-Ansätze [34, 98, 177] oder Strahlschnitttests zwischen jedem modellierten Oberflächenpunkt und allen Flächenelementen der Rohlingoberfläche [33, 69, 92, 177] an. Teilweise werden diese Verfahren durch Ausnutzung

stets identischer geometrischer Bezüge zwischen der Düse und der Auffangfläche algorithmisch vereinfacht [33, 69]. Der durch die Sichtbarkeitsberechnung entstehende Berechnungsaufwand wird dabei als erheblich eingestuft [33, 69]. Wie auch bei den Publikationen im Bereich der thermischen Spritztechnik sind die Verfahrenswege der Düse relativ zum Substrat sowie die Substratform stark eingeschränkt. Dies ist im Aufbau der Sprühkompaktieranlagen begründet.

Erzeugung, Digitalisierung und Nachbearbeitung von Footprintprofilen

Grundlage für jede Simulation des thermischen Beschichtens mit dem Ziel, die auf einem Bauteil abgelagerte Materialverteilung zu berechnen, ist eine Beschreibung des vom Spritzbrenner in einer bestimmten Zeitspanne richtungsabhängig abgegebenen Materials. Für die im Rahmen dieser Arbeit entwickelte Beschichtungssimulation, welche in den Kapiteln 4–6 präsentiert wird, basiert diese Beschreibung auf einem für den zu simulierenden Spritzprozess repräsentativen Footprintprofil, im Folgenden auch kurz *Footprint* genannt.

Definition. *Unter einem Footprint \mathcal{F} wird dasjenige dreidimensionale Profil des beim thermischen Spritzen abgelagerten Spritzmaterials verstanden, welches entsteht, indem unter definiertem (Raum-)Winkel, Abstand und für eine definierte Zeit auf ein initial ebenes, relativ zur Spritzpistole unbeweglich positioniertes Substrat gespritzt wird.*

Der Raumwinkel, unter dem bei einem Footprintexperiment auf die Oberfläche gespritzt wird, wird als *Spritzwinkel* bezeichnet und klassischerweise zwischen der ebenen Oberfläche und der durch die Ausrichtung des Brenners gegebenen Spritzachse gemessen. Da dieser Winkel bei nichtplanaren Oberflächen nicht eindeutig zu bestimmen ist, wird der Spritzwinkel in dieser Arbeit entgegen der klassischen Definition *nicht* zur Oberfläche, sondern zur Oberflächennormalen gemessen. Sofern nicht anderes angegeben ist, beträgt der Spritzwinkel für die im Rahmen dieser Arbeit durchgeführten Footprintexperimente 0° und die Form des Footprintprofils gibt damit den Materialfluss in einer senkrecht zu Spritzrichtung ausgerichteten Querschnittsebene durch den Spritzstrahl wieder. Abbildung 3.1 zeigt die Fertigung von Footprints mit dem Lichtbogenspritzverfahren sowie dem HVOF-Verfahren.

Die Form eines Footprints ist nicht nur charakteristisch für eine Spritztechnologie, wie zum Beispiel Lichtbogenspritzen oder HVOF, sondern auch für die Parametrisierung jedes Prozes-

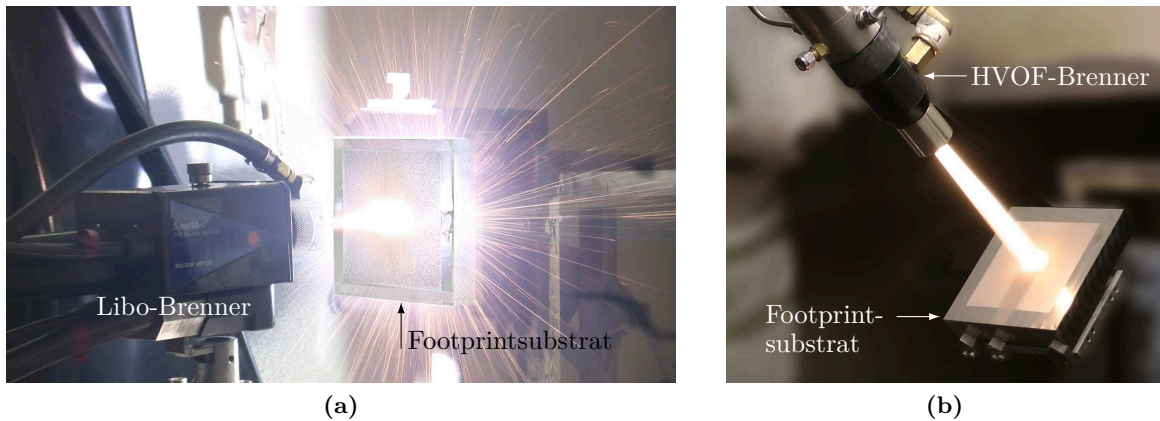


Abbildung 3.1: Fertigung von Footprints (a) mit dem Lichtbogenspritzprozess und (b) mit einem HVOF-Prozess.

ses durch die Wahl der Maschinenparameter und das verwendete Spritzmaterial. Eine möglichst genaue messtechnische Erfassung des Footprintprofils für den zu simulierenden Prozess ist daher maßgeblich für die Realisierung einer exakten Beschichtungssimulation. In diesem Kapitel werden daher detailliert die auf Basis einer Vielzahl von Versuchen entwickelten technischen und algorithmischen Vorgehensweisen zur Erzeugung, Digitalisierung und Nachbearbeitung der im Rahmen dieser Arbeit verwendeten Footprintprofile erläutert.

Im folgenden Abschnitt 3.1 wird zunächst ein Überblick über den Stand der Forschung und Technik gegeben, wobei insbesondere verschiedene Methoden zur Bestimmung von Footprintprofilen beziehungsweise der Spritzcharakteristik sowie unterschiedliche messtechnische Verfahren betrachtet werden. Im Anschluss wird in Kapitel 3.2 eine Übersicht über die gewählte Verfahrensweise zur Herstellung, Digitalisierung und Nachbearbeitung von Footprints gegeben, welche in den darauffolgenden Kapiteln 3.3 bis 3.9 soweit möglich in chronologischer Reihenfolge der Einzelschritte erläutert wird. Im Fokus stehen dabei primär die Verfahrensschritte, welche im Hinblick auf die Verwendung der resultierenden Profildaten in der entwickelten Beschichtungssimulation von Bedeutung sind. Diese umfassen sowohl relevante Aspekte von Standardvorgehensweisen als auch im Rahmen dieser Arbeit entwickelte technische und algorithmische Verbesserungen. Die Gesamtheit der im Rahmen dieser Arbeit entwickelten Verfahren verfolgt dabei das Ziel, die Robustheit des technischen Prozesses und die Messgenauigkeit der Footprintprofile durch die Vermeidung und Reduzierung von Störeinflüssen zu verbessern.

3.1 Stand der Forschung und Technik

Konkrete Verfahrensweisen, welche die Fertigung, Digitalisierung und gegebenenfalls Nachbearbeitung dreidimensionaler Footprintprofile, die in der Literatur auch als „spray spots“, „spray patterns“ oder „deposition patterns“ bezeichnet werden, umfassend behandeln, sind in

der Literatur – mit Ausnahme der im Rahmen dieser Arbeit aus diesem Anlass entstandenen Veröffentlichung [171] – nach Kenntnis des Autors nicht vorhanden. Inhaltliche Schnittmengen dieser Thematik ergeben sich jedoch mit Veröffentlichungen aus dem Bereich der thermischen Spritztechnik, in denen Methoden zur Vermessung von Spritzprofilen Anwendung finden, sowie mit Veröffentlichungen aus dem Bereich der allgemeinen und oberflächenorientierten Messtechnik. Darüber hinaus gibt es alternative Ansätze zur Messung der Massenflusscharakteristik im Freistrahls oder zur simulationsbasierten Berechnung des Massenflusses auf Basis von Düsen- und Freistrahlsimulationen. Während in diesem Forschungsgebiet die CFD-Simulation (engl.: Computational Fluid Dynamics) von thermischen Spritzvorgängen sowohl innerhalb verschiedener Düsen in Spritzbrennern als auch außerhalb der Brenner im Freistrahls ein Thema breiten Forschungsinteresses darstellt, wird auf die weitergehende Verwendung der Simulationsergebnisse zur tatsächlichen Beschichtungsberechnung selten eingegangen. Dennoch existieren einige Ansätze zur Kopplung freistrahlsbasierter mit substratbasierten Simulationen.

Remesh et al. [119] verwenden beispielsweise die kommerzielle CFD-Software Fluent¹ zur Berechnung von Temperatur-, Geschwindigkeits- und Partikelgrößenverteilungen im Freistrahls eines Plasmabrenners als Grundlage für eine anschließende Berechnung des Ausbreitungsfaktors verschiedener Splats nach dem Modell von Pasendideh-Fard et al. [113]. Durch die simulierte Ablagerung der auf diese Weise charakterisierten Splats als einfache Zylinder in einem Voxelvolumen konnten sowohl stationäre Footprints als auch Querschnittsprofile eines linear verfahrenen Spritzbrenners simuliert werden, welche qualitative Parameter wie die Streckung der Footprintform in Richtung der Partikeleindüsung im Brenner oder die Flachheit eines Querschnitts in ihrer Tendenz richtig abbilden. Lugscheider et al. [95, 96] simulieren die Partikelverteilung vor dem Substrat auf Basis der Verfolgung von Partikeltrajektorien durch die Domäne einer Freistrahlsimulation und berechnen damit den Schichtauftrag auf komplexen Oberflächen durch das „Aktivieren“ vordefinierter, finiter Elemente auf dem Substrat. Zhang et al. [181] betrachten ebenfalls die Trajektorien einzelner Partikel verschiedener Größe im Freistrahls und entwickelten hierfür ein probabilistisches Lattice-Boltzmann-Verfahren (LB-Verfahren) basierend auf hexagonalen Gittern. Die auf die Partikel wirkenden Kräfte werden dabei auf Grundlage einer LB-basierten Simulation des Freistrahls berechnet. Kang et al. [76] verwenden eine Fluent-basierte Freistrahlsimulation und ein einfaches Mikrostrukturmodell auf Basis von zylindrischen und unter flacheren Spritzwinkeln elliptischen Splatformen konstanter Höhe, um Footprintgeometrien unter verschiedenen Spritzwinkeln zu generieren. Diesem Grundprinzip folgend können theoretisch beliebige Freistrahlsimulationen, welche die für die Berechnung der Partikeltrajektorien notwendigen Feldgrößen bereitstellen, zur simulativen Ermittlung eines Footprintprofils herangezogen werden.

Als primäre Grundlage für eine makroskopische Beschichtungssimulation erscheint dieser rein simulative Weg zur Erzeugung einer Footprintgeometrie jedoch derzeit noch wenig geeignet. Dies ist darin begründet, dass der Übergang von den in der Freistrahlsimulation berechne-

¹ANSYS Fluent, <http://www.ansys.com>

ten Feldgrößen zu einer realistischen Mikrostruktur unter Zuhilfenahme einer Einzelpartikelbasierten Simulation sehr schwierig ist. Es kann beispielsweise nicht davon ausgegangen werden, dass der gesamte, über die Feldgrößen einer Freistrahlsimulation bestimmte Massenfluss auf die Mikrostruktur übertragen wird, da abhängig von Auftreffwinkel und Aufschmelzgrad der Partikel unterschiedlich viele Partikel von der Oberfläche abprallen. Ebenso hat der ortsabhängige, in der Mikrostruktur enthaltene Porenanteil einen wesentlichen Einfluss auf die Form und das Gesamtvolumen der abgelagerten Schicht und müsste daher in das Simulationsmodell für den Schichtaufbau einbezogen werden. Aufgrund der sehr hohen Anzahl an Partikeln, die für den Aufbau schon eines sehr kleinen Ausschnitts einer Beschichtung erforderlich sind, und der damit verbundenen hohen Rechenzeit werden hierfür jedoch stark abstrahierte Modelle verwendet, welche einzelne Splats oftmals als ideal zylindrisch annehmen und auf komplexe Interaktionen zwischen Splat und Substrat verzichten [63, 76]. Die Porosität wird in solchen Modellen in der Regel durch einen einfachen, mehr oder weniger vordefinierten Effekt, wie zum Beispiel das uniforme Hochbiegen der Ränder aller Splats als Funktion des Splatradius (Curl-Up-Effekt) [28, 63], modelliert oder auch gänzlich vernachlässigt [76]. Die genaue Form der Poren und die aus der Simulation resultierende Gesamtporosität sind somit bereits im Vorhinein durch die Wahl der Splatformen und eines Curl-Up-Winkels festgelegt. Ein heuristischer Ansatz zur Simulation von Poren in Abhängigkeit der lokalen Eingriffssituation, insbesondere zur Porenbildung zwischen aneinander angrenzenden oder aufgebrochenen Splats, auf Basis digitaler Filterung der simulierten Beschichtungsstrukturen wird in [83] vorgestellt. In [172] stellt der Autor dieser Dissertation ein effizientes Simulationsverfahren für den Mikrostrukturaufbau basierend auf einer GPU-beschleunigten Einzelpartikelsimulation vor, welches in der Lage ist, den dreidimensionalen Schichtaufbau aus mehr als 10.000 Einzelpartikelablagerungen unter Berücksichtigung der Aufprallsituation jedes einzelnen Partikels zu simulieren. Trotz der durch den Einsatz effizienter Implementierungstechniken erreichten Erweiterung der Grenzen modellierbarer Mikrostrukturgefüge reicht die heute zur Verfügung stehende Rechenleistung jedoch nicht für eine hinreichend detaillierte Simulation des Schichtaufbaus eines vollständigen Footprintprofils aus. Insgesamt ist es daher nicht überraschend, dass in der Literatur zur simulativen Footprinterzeugung keine aussagekräftigen Daten zur quantitativen Genauigkeit der simulativ ermittelten Profile zu finden sind. Lediglich Kang et al. [76] stellen das Volumen und die Maximaldicke der simulativ und experimentell ermittelten Footprintprofile direkt gegenüber. Allerdings sind diese Daten nicht stichhaltig, da Abweichungen von bis zu 27% im Volumen vollständig und ohne messtechnische Überprüfung der nicht in der Simulation berücksichtigten Porosität zugeordnet werden.

Abgesehen von diesen Schwierigkeiten der quantitativen Validierung wird für eine realistische CFD-basierte Footprinterzeugung neben der genauen Geometrie der Brennkammer bzw. der Düse eine enorme Vielzahl an prozess- und werkstoffspezifischen Variablen und Konstanten benötigt, welche im Allgemeinen nicht ohne Weiteres zur Verfügung stehen. Dazu zählen mechanische, chemische, thermische und elektrische Materialkennwerte, welche nicht nur bei Raumtemperatur sondern in vielen Fällen für einen weiten Temperaturbereich bekannt sein müssen. Für eine vollständige Simulation von der Brennkammer bis zum Erkalten der Schicht

auf dem Substrat sind solche Materialeigenschaften nicht nur für das Spritzgut, sondern auch für die Elektroden in der Brennkammer, den zu beschichtenden Werkstoff und umgebende Gase von Bedeutung. Darüber hinaus müssen gegebenenfalls das konkrete Bauteil, seine Oberflächenbeschaffenheit und die Ausrichtung und Positionierung des Brenners modelliert, in eine zur Berechnung geeignete FE-Zerlegung unterteilt sowie korrekte Randbedingungen (z.B. Temperaturen, Drücke) definiert werden. In der Praxis würde daher die korrekte Einstellung der kombinierten Freistrahlmikrostruktur-Simulation auf einen spezifischen Prozess auf einer spezifischen Anlage ein hohes Maß an Expertenwissen in Bezug auf die Simulation voraussetzen und gegebenenfalls eine zusätzliche Überprüfung der Zwischenergebnisse der Freistrahlsimulation erfordern.

Bei Ermittlung eines Footprintprofils durch Messungen am realen Prozess kann aus theoretischer Sicht zwischen Messungen im Spritzstrahl und der Vermessung des abgelagerten Werkstoffs unterschieden werden. Für Messungen im Spritzstrahl können kommerzielle Sensoren wie z.B. die Systeme AccuraSpray, Sprayview oder DPV-2000 von Tecnar², SprayWatch von Oseir³ oder Linspray PFI (Particle Flux Imaging) von Linde Gas⁴ eingesetzt werden, welche Partikeldichten, -größen, -temperaturen und -geschwindigkeiten messen. Je nach System werden dabei entweder stark lokalisierte Einzelpartikelmessungen in einem kleinen Messbereich ($< 1 \text{ mm}^3$) oder aber Messungen in einem größeren Bereich bis hin zum gesamten Spritzstrahl durchgeführt, welche dann allerdings in der Regel nur angenäherte Durchschnittswerte liefern [57]. Zu diesen Einschränkungen kommt erschwerend hinzu, dass verlässliche Messungen häufig nur bei Prozesseinstellungen mit geringerer Partikeldichte und nicht zu nah an der Brennerdüse erreicht werden können, da andernfalls, insbesondere bei den für die Footprintmessung relevanten Sensoren mit großem Messfeld, eine Sensorüberlastung aufgrund der zu hohen Strahlungsintensität der Partikel oder des Plasmas auftreten kann [101]. Ähnliche Probleme treten aus bislang ungeklärter Ursache bei bestimmten Werkstoffen (z.B. Mullit, Lanthan-Hafnat) auf. Gevelber et al. [61] haben gezeigt, dass es zwar möglich ist, Strahlsensorik gewinnbringend für eine closed-loop-Regelung des Prozesses zur Einhaltung vorgegebener Partikeltemperaturen anzuwenden, jedoch konnte keine aussagekräftige Relation zwischen den Sensordaten und der erzeugten Schichtdicke hergestellt werden. Insgesamt werden durch die Messung im Strahl im Vergleich zur rein simulativen Bestimmung der Footprints zwar die Unsicherheiten durch eine potentiell ungenaue Freistrahlsimulation mit eventuell vorgeschalteter Brennkammersimulation eliminiert, selbst unter der Annahme ideal arbeitender Sensorik verbleibt jedoch auch hier die Problematik, die gemessenen Freistrahlgößen korrekt in Footprintabmessungen auf dem Substrat zu überführen. Daher sind auch diese „In-Flight“-Messungen in der Praxis – zumindest noch – nicht in der Lage, ein Footprintprofil beziehungsweise einen genau lokalisierten Massenfluss quantitativ verlässlich zu ermitteln.

Abbildung 3.2 verdeutlicht diesen Zusammenhang: Je mehr Teilprozesse des Gesamtprozesses simuliert werden, desto komplexer und damit potentiell auch ungenauer wird die Berechnung

²<http://www.tecnar.com>

³<http://www.oseir.com>

⁴<http://www.linde-gas.com>

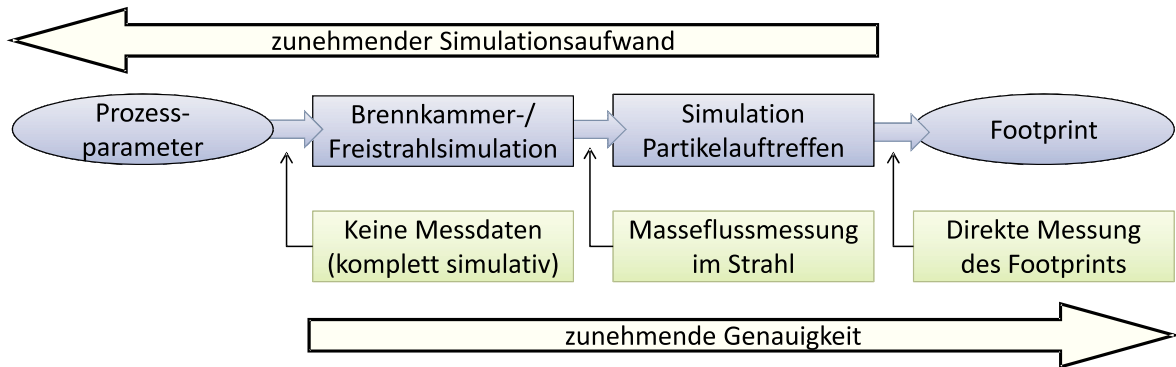


Abbildung 3.2: Rein simulativ kann ein Footprint theoretisch durch die Hintereinanderschaltung von Simulationen für alle Teilprozesse berechnet werden (blau). Werden die zwischen den einzelnen Teilsimulationen weitergegebenen Daten durch Messungen ermittelt, können diese Messdaten anstelle der vorherigen Simulation(en) verwendet werden und die Pfeile der grünen Kästen markieren dann den jeweiligen Einstiegsplatz in der Simulationskette. Im Fall der direkten Vermessung der Footprints ist keine Simulation mehr erforderlich.

des Footprints. Ein durchgehend simulierter Gesamtprozess, welcher ausschließlich auf an der Maschine einzustellenden Prozessparametern basiert, wäre in der Praxis nur mit erheblichem Wissen über die interne Funktionsweise der Simulationen sowie genaue Materialkennwerte der eingesetzten Werkstoffe korrekt einstellbar. Während bei der Materialflussmessung im Strahl noch der schwierige Schritt der Partikelablagerung und Schichtbildung auf Basis der erhaltenen Messwerte simuliert werden muss, kommt die direkte Vermessung des experimentell erzeugten Footprints gänzlich ohne Simulation aus. Auch wenn der dabei entstehende Messfehler nicht direkt mit dem kombinierten Fehler aus der Messung im Strahl und anschließender Simulation der Partikelablagerung verglichen werden kann, so ist die direkte Vermessung des auf einer initial flachen Substratoberfläche abgelagerten Werkstoffs grundsätzlich wesentlich unkomplizierter, da die Messung nicht online, das heißt nicht im laufenden Prozess, durchgeführt werden muss. Für eine Messung steht damit beliebig viel Zeit zur Verfügung, so dass eine Beschränkung auf einen kleinen Ausschnitt des Spritzstrahlquerschnitts respektive der Footprintgrundfläche nicht erforderlich ist und überdies auch mehrere verschiedene Messverfahren nacheinander eingesetzt werden können.

Zur Vermessung von Profilen thermisch gespritzter Schichten kommen sowohl zerstörungsfreie Methoden, beispielsweise basierend auf optischer oder taktiler Messtechnik, als auch zerstörende Methoden, wie die mikroskopische Vermessung in einem oder mehreren Querschliffen der Beschichtung, in Frage. Letztere stellt für die metallographische Untersuchung thermisch gespritzter Proben die Standardtechnik dar [163]. Dabei wird das Substrat mitsamt der anhaftenden Beschichtung entlang einer oder mehrerer senkrecht zur Oberfläche ausgerichteten Ebenen zerteilt, zur besseren Handhabung in eine Einbettmasse eingebettet, geschliffen und poliert und anschließend im Mikroskop analysiert. Durch die Anwendung von Bildanalyseverfahren auf digitale Aufnahmen der mikroskopierten Beschichtung lassen sich eine Vielzahl von Eigenschaften analysieren [4]. Für eine Vermessung der Schichtdicke werden hierzu in

der Regel einige diskrete Messstrecken händisch in einer Messsoftware definiert. Durch das Zusammensetzen mehrerer Einzelaufnahmen zu einem Bild lassen sich Schichtdickenverläufe mit mehreren Zentimetern Gesamtlänge vermessen. Aufgrund hochauflösender Optiken und Bildsensoren können extrem hohe Messauflösungen im Submikrometerbereich einfach erreicht werden [4]. Des Weiteren erlaubt diese Technik im Gegensatz zu nichtinvasiven Scantechnologien die Erfassung und Berücksichtigung des exakten Substratverlaufs beziehungsweise des Verlaufs der Unterseite der Beschichtung. Nachteile des Verfahrens stellen neben der Zerstörung der Probe die sehr zeit- und kostenintensive Präparation und Vermessung der Probe sowie die lediglich zweidimensionale Vermessung dar.

Weitere Verfahren zur direkten Messung der Schichtdicke, welche auch im Bereich thermisch gespritzter Schichten eingesetzt werden, stellen Ultraschall- [120, 185], Wirbelstrom- [51, 107] und Röntgenfluoreszenz-basierte (engl.: X-Ray Fluorescence, XRF) [120] Methoden dar. Diese nichtdestruktiven Methoden beruhen auf einer punktuellen Ultraschallanregung, magnetischen Induzierung von Wirbelströmen oder Röntgenbestrahlung und der darauffolgenden Messung der an Grenzschichten reflektierten Schallwellen, der Impedanz oder der Absorption und Dämpfung der Röntgenfluoreszenz des Substrats durch die Beschichtung. Die Durchführung dieser Messungen erfolgt jedoch in der Regel als Einzelpunktmessung mittels eines auf die Beschichtung aufgesetzten Handmessgeräts [104]. Im industriellen Einsatz existieren zwar robotergestützte Systeme zur Führung derartiger Sensoren, beispielsweise das System „Robotic PELT“ der Firma Imaginant⁵, die Charakteristik der Messung bleibt dabei jedoch punktbasiert. Für die flächige Profilerfassung sind diese Geräte und Sensoren jedoch ungeeignet.

Im Gegensatz zu den bisher genannten direkten Methoden setzen taktile oder optische Systeme zur Vermessung der äußeren Bauteilform voraus, dass die unterliegende Substratgeometrie, möglicherweise durch eine vorherige Vermessung, bekannt ist und als Referenzgeometrie zur Bestimmung der Schichtdicke herangezogen werden kann. Allgemein existieren zur Vermessung freigeformter Bauteile eine Vielzahl von Verfahren und kommerzieller Systeme, welche für verschiedene Anwendungen unterschiedlich gut geeignet sind. Savio et al. [126] bieten einen guten Überblick über verfügbare Verfahren und bewerten ihre Eignung für diverse Szenarien. Als Kriterien zur Kategorisierung der Anwendungsfälle dienen hierbei die Probengröße und -komplexität, die Oberflächen- und Materialbeschaffenheit sowie die zu erzielende Genauigkeit. Hierbei schränken Oberflächen- und Materialbeschaffenheit die Eignung verschiedener Sensortypen für einen Anwendungsfall ein. So sind beispielsweise taktile Scanner nur bedingt für deformierbare Materialien geeignet. Spiegelnd reflektierende oder transparente Oberflächen können hingegen bei optischen Sensoren Probleme verursachen. Prinzipiell zur Vermessung von Schichtprofilen geeignete und eingesetzte Systeme umfassen unter anderem Koordinatenmessmaschinen (engl.: Coordinate Measuring Machines, CMM) mit taktilem oder optischem Messkopf, Laserscanner, Streifenlichtscanner und Profilometer auf Basis des Fokusvariationsprinzips [37]. Koordinatenmessmaschinen sind aufgrund ihrer hohen Präzision

⁵http://www.imaginant.com/pelt_robopelt.html

und Flexibilität weit verbreitet [57] und werden sowohl mit taktilen als auch laserbasierten Messköpfen zur Vermessung thermisch gespritzter Beschichtungen eingesetzt [35, 76, 131, 158, 159], wobei lediglich Kang et al. [76] vollständig dreidimensionale Vermessungen von Footprints vornehmen und sich nicht auf Querschnittsprofile entlang einer Grundlinie beschränken. In Bezug auf die erzielbare Genauigkeit ist hier kein großer Unterschied zwischen optischen und taktilen Messköpfen zu erwarten – Trifa et al. haben gezeigt, dass die Ergebnisse taktiler und laserbasierter Messungen gut übereinstimmen, wenn berücksichtigt wird, dass taktile Verfahren aufgrund des nicht zu vernachlässigenden Durchmessers des Tastkopfes, welcher nicht in beliebig kleine Profiltäler eindringen kann, etwas höhere Schichtdicken messen [160]. Der Unterschied liegt dabei in der Größenordnung der Rauheit der untersuchten Schicht. Aufgrund der berührungslosen Messung sind optische Methoden in der Regel schneller als taktile und bergen nicht die Gefahr, dass sich die Probe durch die Krafteinwirkung des Mess-tasters verformt oder verschiebt. Letzteres stellt jedoch bei der Vermessung von Footprintprofilen auf Stahlsubstraten, wie sie im Rahmen dieser Arbeit durchgeführt wurden, aufgrund des hohen Probengewichts und der geringen Antastkräfte kein Problem dar. Die beschränkte Eignung optischer Verfahren für spiegelnde Flächen hingegen kann zu Schwierigkeiten im Rahmen einer initialen Referenzvermessung unbeschichteter Bauteile oder Rand- und Seitenflächen der Proben führen.

Die Eignung verschiedener kommerzieller Scansysteme wird in Kapitel 3.5.1 weiterführend diskutiert. Eine umfassende Übersicht über die genannten sowie weitere Techniken zur Vermessung freigeformter Bauteile findet sich in [57]. Schwenke et al. [129] und Weckenmann et al. [166] diskutieren detailliert den Stand der Technik in den Bereichen optischer und taktiler Sensorik.

3.2 Übersicht

Abbildung 3.3 zeigt eine Übersicht der in diesem Kapitel beschriebenen Schritte zur Herstellung, Digitalisierung und Nachbearbeitung von Footprintprofilen in Form eines chronologischen Ablaufplans, welcher die Einordnung einzelner Schritte in den Gesamtprozess erleichtert. Die übergeordneten Ablaufschritte werden in den folgenden, in der Abbildung angegebenen Kapiteln 3.3 – 3.9 behandelt.

Der Gesamtprozess beginnt im Folgenden mit der Probenvorbereitung, welche die Fertigung der Substratproben vom Schneid- und Fräsprozess bis zum Vortemperieren der Proben unmittelbar vor der Beschichtung durch den thermischen Spritzprozess umfasst. Kapitel 3.4 zur robotergestützten Footprintbeschichtung behandelt im Anschluss grundlegende Optionen zur experimentellen Footprintherstellung und thematisiert die für die nachfolgende Vermessung der Profile kritische Einmessung der Proben sowie die thermische Belastung des Schicht-Substrat-Verbunds. Für die Vermessung der erzeugten Profile wird nachfolgend die Eignung von sechs im Rahmen dieser Arbeit erprobten Scansystemen zur Digitalisierung der Footprintprofile diskutiert und die Durchführung des Scanvorgangs mit dem gewählten Gerät

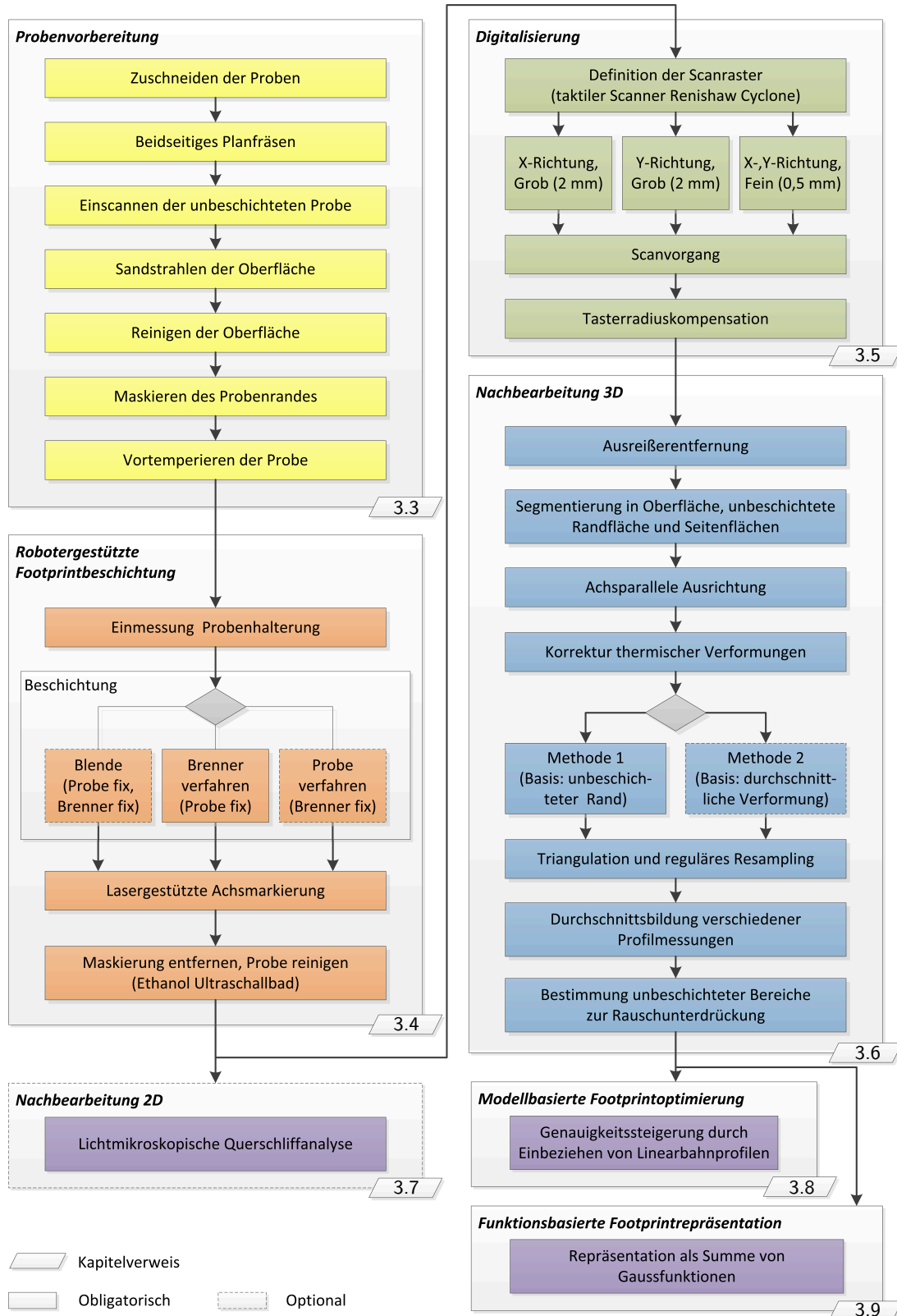


Abbildung 3.3: Gesamtablauf bei der Herstellung, Digitalisierung und Nachbearbeitung von Footprintprofilen.

beschrieben (Kapitel 3.5). Kapitel 3.6 befasst sich schließlich mit der Nachbearbeitung der gewonnenen Scandaten, wobei der Fokus auf der Analyse und der Korrektur thermisch bedingter Verformungen des Schicht-Substrat-Verbunds liegt. Zur Beurteilung der zwei zur Korrektur der Verformung entwickelten Methoden sowie der im taktilen Scanprozess ermittelten Schichtdickenverteilungen auf komplexen Werkstückgeometrien in Kapitel 3.6 wird im Rahmen dieser Arbeit auf Schichtdickenmessungen in lichtmikroskopischen Querschliffbildern zurückgegriffen. Der im Rahmen dieser Arbeit entwickelte Algorithmus zur automatisierten Bestimmung des Schichtdickenverlaufs in Querschliffbildern wird in Kapitel 3.7 dargestellt.

Nach der Nachbearbeitung der dreidimensionalen Scandaten können die ermittelten Profile im Rahmen der Beschichtungssimulation zur Charakterisierung des Massenflusses in Form einer Lookup-Tabelle zum Einsatz kommen. Um die Genauigkeit dieser Repräsentation weiter zu verbessern, kann darüber hinaus die in Kapitel 3.8 vorgestellte modellbasierte Footprintoptimierung eingesetzt werden. Diese nutzt zusätzliche Schichtprofile, welche mit einem geradlinigen Bahnverlauf der Spritzpistole erzeugt wurden, für eine Verbesserung des gemessenen Footprintprofils.

In Kapitel 3.9 wird schließlich die funktionsbasierte Repräsentation gemessener Footprintprofile auf Basis von Summen bivariater Gaußverteilungen beschrieben, welche sowohl eine Alternative zu der Repräsentation als Lookup-Tabelle als auch einen erweiterten Schritt zur Filterung und Interpretation der Profildaten darstellt und sich als solcher in die Kette der Nachbearbeitungsschritte dreidimensionaler Profile eingliedern kann.

Zur Demonstration der Footprinterzeugung, -digitalisierung und -nachbearbeitung wurden in den Kapiteln 3.3 bis 3.9 die Profildaten zweier unterschiedlicher Messreihen verwendet. Die erste Messreihe, bestehend aus mehreren Footprints sowie einigen in Kapitel 3.8 verwendeten Linearbahnprofilen, kommt in den Kapiteln 3.3 bis 3.8 zum Einsatz. Die zweite Messreihe von Footprintprofilen wird für die Auswertungen in Kapitel 3.9 sowie für alle Versuche innerhalb des Evaluationskapitels 6 verwendet. Diese Messreihe wurde auf die gleiche Art und Weise hergestellt, digitalisiert und der entwickelten Nachbearbeitung unterzogen wie die erste Messreihe. Die einzige Ausnahme stellt hier die aufgrund der nicht verfügbaren Linearbahnprofile für die zweite Messreihe nicht durchgeführte modellbasierte Footprintoptimierung dar. Begründet ist die Durchführung einer zweiten Messreihe in der technischen Aufrüstung der für die Speisung der Spritzanlage eingesetzten Druckluftanlage nach der Feststellung, dass die Leistungsfähigkeit der ursprünglichen Anlage nicht ausreichend war, um die eingestellten Primär- und Sekundärgasdrücke dauerhaft im Prozess sicherzustellen. Durch diese für den Autor dieser Arbeit unvorhersehbare, bauliche Maßnahme wurde die Massenflusscharakteristik des verwendeten Lichtbogenspritzprozesses in der Tat dauerhaft in Richtung eines stärker eingeschnürten Spritzstrahls und eines insgesamt runderen Footprintprofils verändert, wie ein Vergleich der Abbildungen 3.34 des alten und 3.39 des neuen Profils zeigt. Die zweite Messreihe diente daher der Ermittlung der Massenflusscharakteristik, welche auch für die Evaluation der Beschichtungssimulation anhand komplexer Bauteile in Kapitel 6 Anwendung fand. Die Interpretation oder Anwendbarkeit der im Folgenden dargestellten Ergebnissen bleibt davon unberührt.



Abbildung 3.4: Stahlsubstrat der Größe $150 \times 150 \times 30 \text{ mm}^3$, dessen umlaufender Randbereich durch angebrachte Blechwinkel vor dem Sandstrahlprozess abgeschirmt wird. Der Randbereich dient später als Referenzfläche für die Digitalisierung der Probe.

3.3 Probenvorbereitung

Als Substrat für die Footprintherstellung dienen Stahlplatten der Größe $150 \times 150 \times 30 \text{ mm}^3$, welche zur Probenvorbereitung zunächst von größeren Stahlstäben abgetrennt und beidseitig plangefräst werden, um eine ebene Oberfläche zu erhalten. Die beidseitige Fräsbearbeitung dient dabei der Vermeidung eines unsymmetrischen Spannungszustandes, welcher durch eine einseitige Bearbeitung entstehen und eine Wölbung der Proben verursachen könnte. Zur Analyse der Planarität, sowie als Referenz für die in Kapitel 3.6.3 beschriebene Korrektur thermischer Verformungen, werden die Substrate bereits vor der Beschichtung wie in Kapitel 3.5 beschrieben eingescannt.

Im folgenden Bearbeitungsschritt werden die Proben im Sandstrahlprozess mit Edel-Korund gestrahlt, um eine Aufrauung der Oberfläche und eine bessere Verklammerung zwischen Substrat und Beschichtung und somit eine bessere Schichthaftung zu erzielen. Dabei wird ein etwa 1 cm breiter, umlaufender Randbereich der Probe, welcher im Rahmen der Profilvermessung als Referenzhöhe dient, abgeschirmt (Abbildung 3.4). Das Strahlen der Proben erfolgt dabei durch händisches, zeilenweises Bewegen der Strahldüse über die Probenoberfläche unter einem Winkel von ca. 45° und in einem Abstand von etwa 100 mm. Um eine Oxidation der aufgerauten Oberfläche zu vermeiden, findet dieser Prozess so kurz wie möglich vor dem Beschichtungsprozess statt. Die gestrahlten Proben werden anschließend in einem Ethanol-Ultraschallbad 15 Minuten gereinigt.

Nach der Reinigung wird der 1 cm breite Probenrand jeder Probe maskiert, um das Anhaften von Partikeln im Spritzprozess zu unterbinden und in diesem Bereich eine saubere, vom Scanner fehlerfrei antastbare Referenzfläche zu erhalten. Die Maskierung geschieht dabei durch das Aufkleben von selbstklebender Maskierungsbänder oder durch Bestreichen der Fläche mit einem schnell trocknenden, wasserlöslichen Maskierungsmittel mit einem Pinsel.

Als letzter Schritt der Vorbereitung werden die Proben schließlich in einem Umluftofen auf ca. 120 °C vortemperiert, um die im Prozess resultierenden thermischen Eigenspannungen zu reduzieren.

3.4 Robotergestützte Footprintherstellung

Zur Erzeugung geeigneter Footprints, welche das prozessdefinierende Massenfluss- bzw. -ablagungsverhalten möglichst korrekt wiedergeben, ist es erforderlich, die ebene Substratfläche für eine festgelegte Dauer unbewegt im Spritzstrahl zu platzieren. Ein grundlegendes Problem besteht dabei darin, dass der Lichtbogenspritzprozess, wie andere thermische Spritzprozesse auch, nicht ab dem Einschaltzeitpunkt stabil läuft. Stattdessen treten in der Phase nach dem Einschalten deutliche Schwankungen der Prozessparameterwerte, wie zum Beispiel der eingestellten Stromstärke, der Spannung oder des Gasflusses auf. Daher ist es nicht möglich, das Footprintssubstrat bereits zu Beginn direkt vor der Spritzpistole zu positionieren, da ansonsten diese unrepräsentativen Schwankungen mit in das Messergebnis einfließen würden. Im Folgenden werden verschiedene Möglichkeiten vorgestellt, dieses Problem zu umgehen.

Eine Möglichkeit besteht darin, eine Blende zwischen Spritzpistole und Substrat zu installieren, welche für eine definierte Dauer entfernt wird, sobald der Prozess sich stabilisiert hat. Die zeitweise Entfernung der Blende geschieht dabei idealerweise automatisiert, beispielsweise durch einen Roboter, so dass die Dauer der Blendenöffnung präzise eingestellt werden kann. Ebenso ist es empfehlenswert, die Blende zum Ende der Beschichtungszeit in der gleichen Richtung wieder in den Spritzstrahl zu bewegen, in der sie zu Beginn auch herausbewegt wurde. Dadurch wird während der Bewegung der Blende genau diejenige Seite des Spritzstrahls zuerst abgeschirmt, welche beim Herausbewegen auch zuerst freigegeben wurde. Andernfalls ist die Bewegungszeit der Blende in die Berechnung der Spritzdauer einzubeziehen. Nachteilig bei dieser Vorgehensweise ist, dass die Konstruktion einer robotergeführten Blende relativ aufwendig ist, da diese, während der Prozess noch nicht stabil läuft, einer sehr hohen thermischen Belastung standhalten muss. Weiterhin ist die Dauer bis zum Erreichen der vorgegebenen Prozessparameter und ihrer Stabilisierung nicht vorhersehbar, da diese Parameter manuell justiert werden und dabei nur zeitverzögert auf Eingaben reagieren.

Eine zweite Möglichkeit zur Gewährleistung einer ausreichenden Prozessvorlaufzeit ist die Montage der Spritzpistole auf einen Roboter, welcher diese erst nach der Prozessstabilisierung über dem Substrat positioniert. Ein Nachteil dieser Methode besteht darin, dass das Footprintssubstrat bereits während der Positionierungsbewegung des Roboters beschichtet wird. Für die im Rahmen dieser Arbeit angefertigten Footprints (Abbildung 3.5) mit einer messbaren Footprintbreite von weniger als 100 mm beträgt die theoretische Positionierungszeit bei einer programmierten Bahngeschwindigkeit des Roboters von 1.000 mm/s ca. 1–2 % der Gesamtspritzdauer von 5–10 Sekunden. Des Weiteren ist bei diesem Ansatz darauf zu achten, dass die Steifigkeit des Roboters und sein maximales Handhabungsgewicht ausreichend sind, um die Spritzpistole möglichst schnell auf ihre Zielposition mittig über dem Substrat zu bewe-

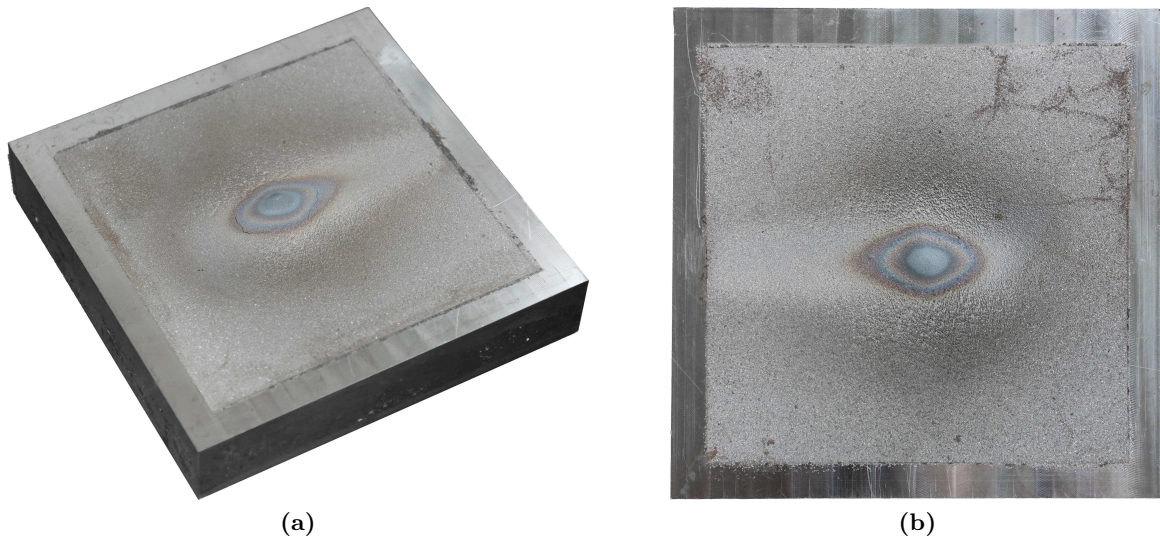


Abbildung 3.5: Zwei Footprintproben, die auf einem Substrat der Größe $150 \times 150 \times 30 \text{mm}^3$ hergestellt wurden. Erkennbar sind der ungestrahlte und unbeschichtete 1 cm breite Rand, welcher als Referenzfläche für die Profilmessung dient, sowie die etwas helleren Beschichtungsspuren, entlang derer der Brenner über dem Substrat platziert wurde.

gen, ohne größere Schwingungen hervorzurufen. Je stärker die Pistole dabei abgebremst wird, desto geringer ist zwar die Positionierungszeit und der damit verbundene Fehler, jedoch steigt auch die Krafteinwirkung auf die Pistole und insbesondere ihre Halterung. Um zu vermeiden, dass sich die Position oder Ausrichtung der Pistole relativ zum Roboter dabei verändert, sollte die Halterung entsprechend stabil und passgenau konstruiert sein. Diese Problematik besteht bei dem zuvor beschriebenen blenden-basierten Ansatz nicht, da Pistole und Substrat ortsfest installiert werden können.

Theoretisch bestünde ferner die Option, das Substrat mit dem Roboter in den Spritzstrahl zu bewegen. Abgesehen von einer erhöhten Gefahr, dass der Roboter in den Spritzstrahl geraten könnte, ist hierfür auch die Konstruktion eines speziellen Probenhalters für den Roboter erforderlich. Dieser müsste die Probe auch bei hohen Beschleunigungen sicher halten, wodurch eventuell eine wesentlich kompliziertere Montage der Substrate erforderlich wird. Für eine nicht auf dem Roboter montierte Probe ist eine einfache Magnethalterung mit zweiseitigem seitlichen Anschlag ausreichend und zudem – im Gegensatz zu einem Roboter – leicht innerhalb oder an einer Abluftanlage zu montieren.

Um die Überhitzungsproblematik des Blenden-basierten Ansatzes und die damit verbundene Konstruktion einer komplexen Blendenvorrichtung zu vermeiden, wurden die im Rahmen dieser Arbeit angefertigten Footprints mittels einer robotergeführten Spritzpistole hergestellt. Dabei kam ein Roboter der Firma ABB⁶ vom Typ 4400/60 mit 60 kg Handhabungsgewicht zum Einsatz, welcher den ohne Schlauchpakete 5,4 kg schweren Brenner mit einer programmierten Geschwindigkeit von 1.000 mm/s auf den Mittelpunkt des Footprintsubstrats bewegt.

⁶ABB Ltd. (Asea Brown Boveri), <http://www.abb.com>

Die durch den Positionierungsweg erzeugte geringe Schichtablagerung kann durch die modellbasierte Footprintoptimierung oder durch die Berücksichtigung im Rahmen der funktionsbasierten Footprintrepräsentation, welche in den Kapiteln 3.8 respektive 3.9 vorgestellt werden, kompensiert werden.

Unabhängig von der Frage, ob nun die Spritzpistole, das Substrat oder die Blende bewegt wird, treten zwei wesentliche Schwierigkeiten bei der Anfertigung der 3D-Footprintprofile auf, welche im Folgenden diskutiert werden: die genaue Einmessung der Probe und des Roboters sowie die mit der hohen thermischen Belastung einhergehende Verformung des Schicht-Substrat-Verbunds.

3.4.1 Einmessung des Werkstücks

Im Rahmen der Simulation des thermischen Spritzvorgangs wird das gewonnene Footprintprofil verwendet, um die in unterschiedlichen Richtungen von der virtuellen Spritzpistole abgestrahlte Materialmenge zu definieren. Dazu muss das gewonnene Footprintprofil in Bezug auf das Koordinatensystem – und damit insbesondere auch in Bezug auf die Quellposition des Massenflusses – dieser virtuellen Spritzpistole korrekt positioniert werden. Um dies zu gewährleisten, müssen alle relevanten Koordinatensysteme von der Spritzumgebung, in welcher der Roboter platziert ist, über die Scannerkoordinatensysteme bei der Digitalisierung des Footprints bis hin zur Simulation nachverfolgt und zueinander in Beziehung gesetzt werden. Grundbedingung hierfür ist eine präzise Einmessung des Roboters, des Werkzeugs und des Werkstücks sowohl bei der Anfertigung der Footprints als auch bei der Durchführung zuvor simulierter Beschichtungen komplexerer Bauteile.

Im Folgenden werden daher die verschiedenen Koordinatensysteme erläutert, welche für die Footprintgewinnung in der angegebenen Reihenfolge zueinander in Bezug gesetzt werden müssen und in Abbildung 3.6 dargestellt sind:

1. Weltkoordinatensystem

Der Ursprung des Welt- oder Basiskoordinatensystems des Roboters wird im Allgemeinen an der Basis des Roboters angenommen und ist mit einer Achse an der ersten Drehachse des Roboterarms ausgerichtet.

2. Flanschkoordinatensystem

Das Flanschkoordinatensystem des Roboters liegt in der Mitte der Flanschplatte, an der das Werkzeug befestigt wird. Die Koordinatentransformation zwischen Weltkoordinatensystem und Flanschkoordinatensystem ist dem Robotersystem aufgrund der Gelenkstellungen bekannt und wird in der Regel intern als kinematische Kette über die Stellung der verschiedenen Gelenkachsen sowie deren Geometrie auf Basis der Denavit-Hartenberg-Transformation [39] berechnet.

3. Werkzeugkoordinatensystem

Das Werkzeugkoordinatensystem hat seinen Ursprung im Arbeitspunkt des Werkzeugs, dem Tool Center Point (TCP). Für den Spritzbrenner ist hierfür der Punkt gewählt

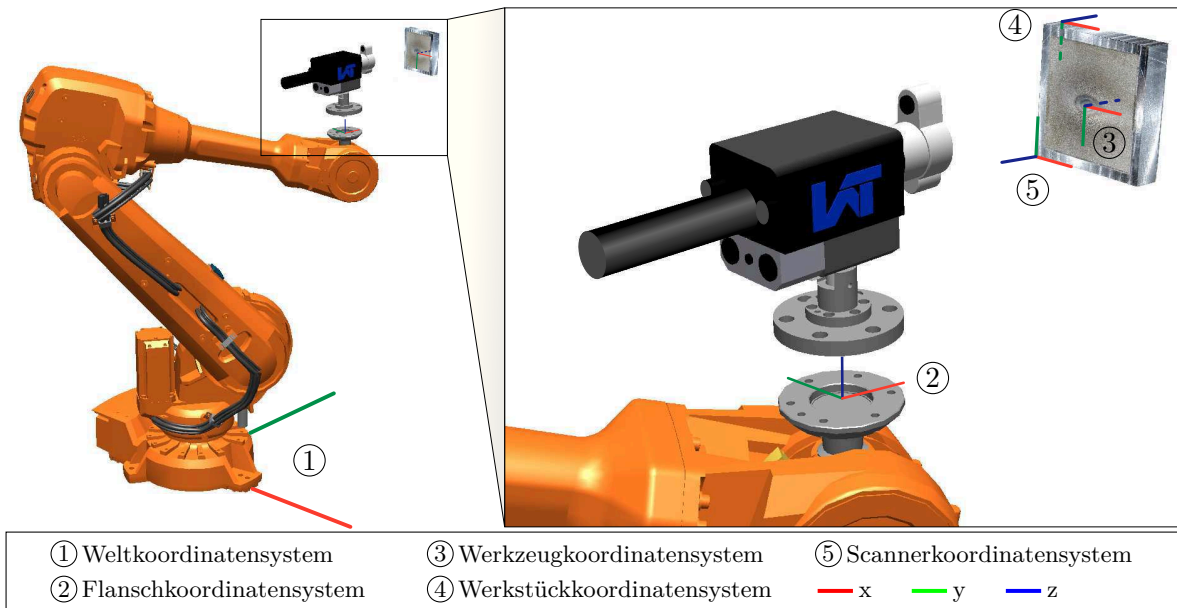


Abbildung 3.6: Übersicht über die verschiedenen Koordinatensysteme. Aufgrund der besseren Lesbarkeit sind die z-Achse des Weltkoordinatensystems, welche von der Roboterbasis ausgehend nach oben gerichtet ist, sowie das Footprintkoordinatensystem nicht dargestellt. Die Achsen des Footprintkoordinatensystems verlaufen parallel beziehungsweise antiparallel zu den x- und y-Achsen des Werkstückkoordinatensystems und sein Ursprung liegt in dem des Werkzeugkoordinatensystems. Der Probenanschlag oben links am Substrat definiert den Ursprung des Werkstückkoordinatensystems.

worden, welcher im optimalen Spritzabstand – 110 mm beim Lichtbogenspritzen – vor der Düse liegt. Dieser Punkt kann prinzipiell anhand der bekannten Geometrie der Spritzpistole und ihrer Halterung relativ zum Flanschkoordinatensystem festgelegt werden. Die Achsen sind dabei so ausgerichtet, dass die z-Achse in Spritzrichtung zeigt und die x- und y-Achsen parallel respektive senkrecht zur Flanschplatte stehen und ein rechtshändiges Koordinatensystem aufspannen.

4. Werkstückkoordinatensystem

Im Koordinatensystem des Werkstücks werden von der Bahnplanungssoftware die Bahnpunkte definiert, entlang derer der TCP im Prozess über die Werkstückoberfläche zu verfahren ist. Dieses Koordinatensystem wird entweder am Werkstück selbst oder häufig an einer Werkstückhalterung, welche einen definierten Anschlagpunkt bereitstellt, um den Werkstückwechsel zu vereinfachen, platziert und mit Hilfe einer am Roboterflansch montierten Einmessspitze durch Anfahren mehrerer Punkte in das Robotersystem einprogrammiert. Für die Footprintgeometrien liegt der Ursprung dieses Koordinatensystems, wie in Abbildung 3.6 gezeigt, an einer der rückseitigen Ecken der quaderförmigen Substrate.

5. Scannerkoordinatensystem

Nach der Herstellung eines Footprints wird dieser mit einem 3D-Scanner vermessen.

Die resultierenden Daten liegen anschließend in Form einer Punktwolke, eines Netzes oder eines Höhenfeldes im Scannerkoordinatensystem vor. Dieses bezieht sich in der Regel auf den Arbeitsbereich des Scanners oder wird während des Scanprozesses in Bezug zur gescannten Probe definiert. Während der Nachbearbeitung der Scandaten wird das Werkstückkoordinatensystem auf Basis der vorliegenden Geometriedaten bestimmt. Für die Footprints substrate liegt der Ursprung an einer der vorderen Eckpunkte des Substrates.

6. Footprintkoordinatensystem

Das Footprintkoordinatensystem beschreibt ein zweidimensionales, in der Simulation verwendetes Koordinatensystem, über dem das in der Simulation verwendete Höhenprofil des Footprints definiert ist. Das Koordinatensystem ist dabei derart in das Werkstückkoordinatensystem eingebettet, dass x- und y-Achsen parallel verlaufen, wobei die y-Achsen jedoch in entgegengesetzte Richtungen zeigen, da sich das Footprintkoordinatensystem aus softwaretechnischen Gründen am Standardkoordinatensystem von OpenGL orientiert. Die Skalierung der Achsen ist in diesem Koordinatensystem so gewählt, dass der für die Simulation relevante Bereich des Footprints den Bereich $(-1, -1)$ bis $(1, 1)$ einnimmt. Der Ursprung des Footprintkoordinatensystems liegt ferner im Schnittpunkt der Werkstückoberfläche mit der Spritzrichtung, welcher für eine im Spritzabstand von 110 mm zur Oberfläche platzierte Probe dem TCP des Lichtbogenspritzbrenners entspricht.

Fehlerquellen bei der Koordinatensystembestimmung

Bei der Bestimmung der verschiedenen Koordinatensysteme haben sich im Rahmen dieser Arbeit zwei Faktoren als besonders problematisch erwiesen: Zum einen die geringe Steifigkeit der Pistole beziehungsweise zu viel Spiel in ihrer nicht ausreichend passgenauen Halterung und zum anderen die Schwierigkeit, eine genaue Spritzrichtung für Footprintversuche und die Beschichtung komplexer Werkstücke wiederholbar festzulegen. Die geringe Steifigkeit und der Spielraum in der Halterung der Spritzpistole sind dabei ursächlich für Abweichungen zwischen Flansch- und Werkzeugkoordinatensystem, welche eine zuverlässige Ausrichtung des Brenners auf das Werkstück beeinträchtigen. Zur Kompensation dadurch auftretender statischer Abweichungen und zur Sicherstellung einer identischen Brennerausrichtung auf das Bauteil über mehrere Beschichtungsvorgänge hinweg ist daher ein manueller Abgleich der Ausrichtung auf Basis eines Spritzbrenner-gebundenen Koordinatensystems notwendig. Wesentlich ist hierfür eine genaue Festlegung der Spritzrichtung des Brenners, welche insbesondere für den verwendeten Lichtbogenspritzprozess nicht auf einfache Weise definierbar ist. Im Folgenden werden diese Probleme zunächst genauer erläutert.

Als Ursache für Fehlausrichtungen des Spritzbrenners wurden zwei wesentliche Faktoren identifiziert. Zum Einen zeigte eine Überprüfung der Pistolenhalterung, welche als Bindeglied zwischen Flanschplatte und Brenner fungiert, dass die Schraubfixierung der Lichtbogenspritzpistole in dieser Halterung nicht ausreichend ist, um ein Verdrehen des Brenners durch die im

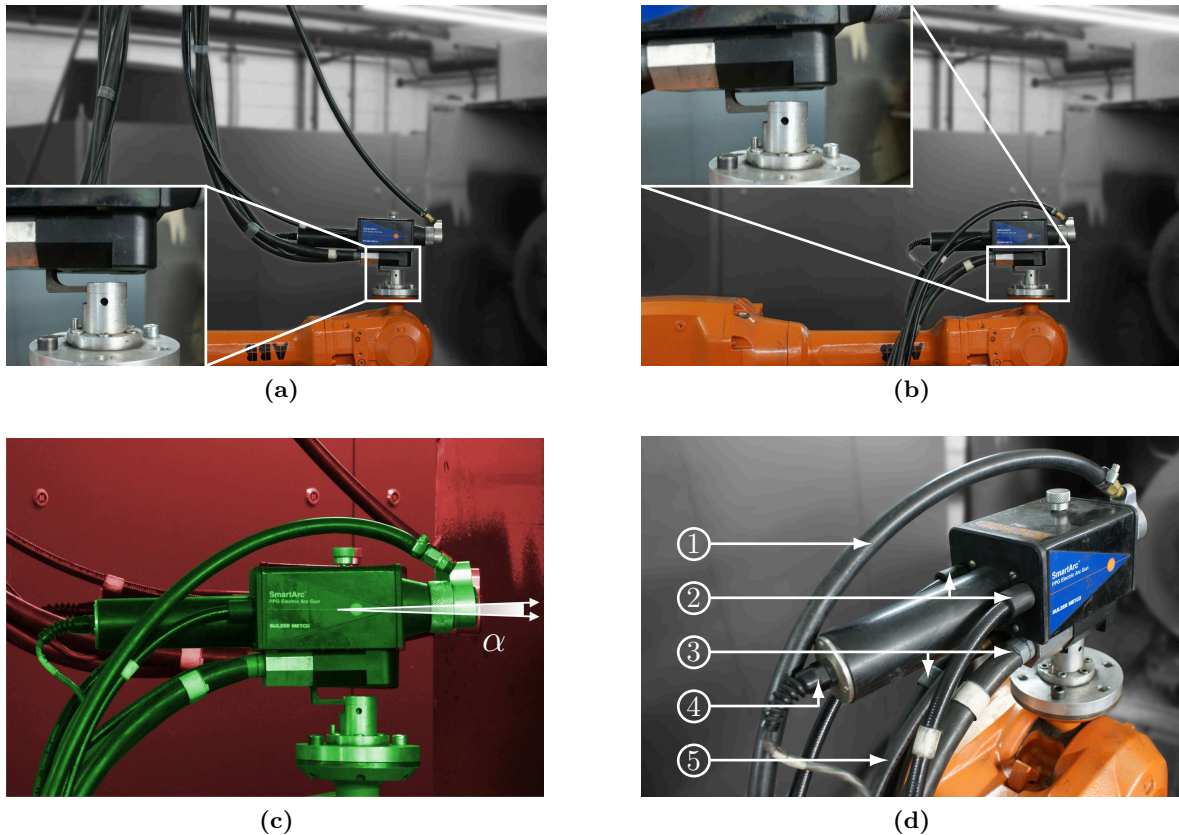


Abbildung 3.7: Zwei unterschiedliche Arten der Schlauchführung: (a) An einem Federsystem an der Decke aufgehängt oder (b) auf dem Boden liegend. (c) Die Überlagerung der beiden Aufnahmen (rot/grün) zeigt, dass die Art der Schlauchführung die Pistole in unterschiedliche Richtungen belastet, was zu einer Neigung der Pistole in diesem Fall um $\alpha \approx 4,25^\circ$ führt. Das Schlauchpaket mit einem Gewicht von ca. 50 kg setzt sich zusammen aus ① Sekundärgaszufuhr, ② Drahtzufuhr (Anode, Kathode), ③ Strom (Anode, Kathode) und Primärgas, ④ Stromversorgung Drahtförderer und ⑤ Druckluftzufuhr Drahtförderer (d).

Prozess wirkenden Beschleunigungen zu unterbinden. Der Brenner kann sich in der nicht exakt passgenauen Halterung dadurch um wenige Grad um die zur Flanschplatte senkrechte z-Achse des Flanschkoordinatensystems (Abbildung 3.6) drehen. Zum Anderen ist die originale Maschinenhalterung der SmartArc-Spritzpistole in Form eines angeschraubten Stahlwinkels nicht steif genug, um eine Verkippung der Pistole um eine zur x-Achse des Werkzeugkoordinatensystems parallele Achse effektiv zu verhindern (Abbildung 3.7). Maßgeblich für den Grad der dadurch hervorgerufenen Fehlansrichtung ist nicht nur das Eigengewicht des Brenners von ca. 5,4 kg, sondern insbesondere auch das Gewicht des an der Pistole befestigten Schlauchpakets (ca. 50 kg, Abbildung 3.7d) sowie die Art der Schlauchführung. Die Schläuche können dabei entweder auf dem Boden entlanglaufen oder über ein Federzugsystem an der Decke geführt werden. Eine Schlauchführung entlang des Roboters kommt hier aufgrund der hohen Biegesteifigkeit und der stärkeren Krümmung, welche die Drahtförderung durch höhere Reibung

erschweren würde, nicht in Frage. Abhängig von der Art der Schlauchführung und der Stellung des Roboters ergibt sich eine am hinteren Ende der Pistole angreifende zeit- oder positionsabhängige Kraft, welche auf die Spritzpistole wirkt und diese aufgrund der erläuterten geringen Steifigkeit der Halterung – wie in Abbildung 3.7 dargestellt – in unterschiedliche Richtungen kippt. Diese Problematik tritt nicht nur bei dem verwendeten Lichtbogenspritzsystem auf, sondern bei allen kabel- bzw. schlauchgebundenen Beschichtungssystemen. Darüber hinaus wirkt der von der Roboterhalterung oftmals weit nach vorn auskragende Brenner schwingungsfördernd – ein typisches Beispiel für einen solchen weit auskragenden Brenner ist der in Abbildung 3.8 dargestellte C-CJS-HVOF Brenner der Firma Thermico⁷.

Diese Gegebenheiten führen dazu, dass das in der Robotersteuerung auf Basis der CAD-Daten der Spritzpistole und ihrer Halterung definierte Werkzeugkoordinatensystem nur eingeschränkt der tatsächlichen Position und Ausrichtung der Pistole entspricht. Grundsätzlich kann dabei zwischen Abweichungen aufgrund von dynamischen Effekten innerhalb eines Beschichtungsvorgangs und Abweichungen, welche auch in quasi-statischen Situationen relevant sind, unterschieden werden. Dynamische Abweichungen zwischen dem programmierten und dem tatsächlichen Werkzeugkoordinatensystem sind insbesondere zu erwarten, wenn die Pistole schnellen Beschleunigungen ausgesetzt ist, welche Werkzeug und Halterung stark belasten. Ein typisches Beispiel ist die abrupte Änderung ihrer Ausrichtung beim Passieren einer starken Werkstückkrümmung bzw. -kante. Demgegenüber stehen Abweichungen der Spritzpistole, welche aus einer Positions- oder Lageänderung der Pistole, des Schlauchpakets oder des Roboters resultieren. Diese treten nicht unbedingt in ausgeprägter Form innerhalb eines Beschichtungsvorgangs auf, sondern ändern sich mit den für verschiedene Werkstücke unterschiedlichen Beschichtungsstrategien, beispielsweise einer geänderten Schlauchaufhängung oder dem Beschichten mit einer um 90° oder 180° um die Spritzrichtung gedrehten Pistole. Im Gegensatz zu den dynamischen Effekten, deren komplexe Simulation ein eigenständiges Forschungsgebiet darstellt und welche daher nicht Teil dieser Arbeit ist, lässt sich eine näherungsweise statische Abweichung des Werkzeugkoordinatensystems durch Messungen der Spritzpistolenlage relativ zum Bauteil ermitteln und durch eine Änderung der Roboterprogrammierung oder -einmessung kompensieren.

Dabei ergibt sich das zweite zu Beginn genannte Problem: die genaue Feststellung der Spritzrichtung, welche die z-Achse des Werkzeugkoordinatensystems definiert und nicht in jedem Fall auf einfache Weise anhand der äußeren Konstruktion des Brenners ersichtlich ist. Eine naheliegende Wahl für die Spritzrichtung und somit die z-Achse des Werkzeugkoordinatensystems ist die Mittelachse der in der Regel rotationssymmetrischen Brennkammer und Düse eines Spritzbrenners. Für den in Abbildung 3.8 dargestellten HVOF-Spritzbrenner ist diese Achse auch in der Praxis anhand der langgezogenen, zylindrischen Geometrie der Spritzpistole gut ersichtlich und eindeutig definiert. Bei Lichtbogenspritzsystemen existiert jedoch eine solche Brennkammer nicht, da die durch den Zerstäubergasstrom von Anode und Kathode mitgerissene Schmelze keine einschnürende Düsengeometrie durchläuft. Um dennoch ein an

⁷Thermico GmbH & Co. KG, <http://www.thermico.de>

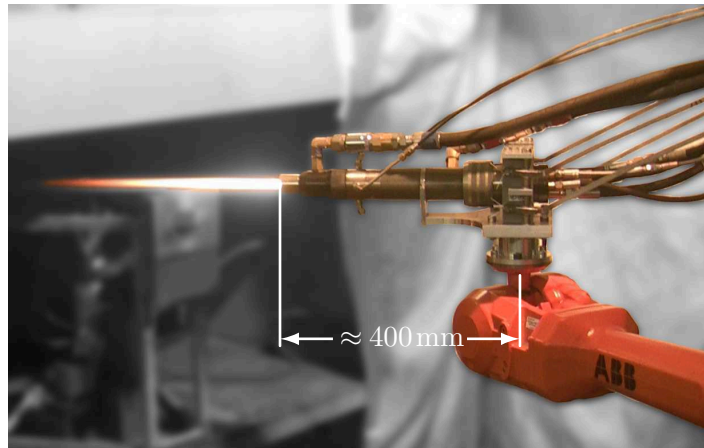


Abbildung 3.8: Der auf dem Roboter montierte und mit ca. 400 mm weit nach vorne auskragende HVOF-Brenner neigt im Betrieb zu nicht unerheblichen Schwingungen (Foto: LWT).

die Spritzpistole gebundenes Werkzeugkoordinatensystem zuverlässig und mit ausreichend hoher Wiederholgenauigkeit über mehrere Versuche hinweg festzulegen, wurde im Rahmen dieser Arbeit eine Lasereinmessvorrichtung entwickelt und – in Kooperation mit dem Lehrstuhl für Werkstofftechnologie – konstruiert und gefertigt.

In diesem Zusammenhang ist es essentiell, zwischen der durch die Lasereinmessung bestimmten, in Spritzrichtung zeigenden Achse des Werkzeugkoordinatensystems, welches sich an der Geometrie des Spritzbrenners orientiert, und derjenigen Richtung \mathbf{q} , in der die überwiegende Menge des Spritzwerkstoffs aus der Pistole abgegeben wird, zu unterscheiden. Das Werkzeugkoordinatensystem dient primär der korrekten Positionierung des Roboters relativ zum Werkstück und stellt das Verbindungsglied zwischen Flansch- und Werkstückkoordinaten dar. Im Idealfall koinzidiert die Massenflussrichtung mit der z -Achse dieses Koordinatensystems, denn nur in diesem Fall kann vom Anstellwinkel der Spritzpistole an einem Bahnpunkt der Roboterbahn zur Oberfläche direkt auf den für die Schichtqualität wichtigen Spritzwinkel, also den Winkel unter dem der Massenfluss die Bauteiloberfläche trifft, geschlossen werden. In der Realität weicht die mittlere Flugrichtung der Partikel jedoch in unterschiedlichem Maße von dieser Achse ab. Dies ist darin begründet, dass die Richtung des größten Massenflusses von den eingestellten Prozessparametern abhängig ist und beispielsweise beim Lichtbogenspritzverfahren durch das ungleichmäßige Abschmelzen von Anode und Kathode [165] sowie die Verwendung unterschiedlicher Düsenaufsätze [184] oder bei pulverbasierten Verfahren wie dem Plasmaspritzen durch die Pulverfraktion, den Trägergasvolumenstrom und die Positionierung und Ausrichtung der Pulverinjektoren des Spritzbrenners [55, 140] beeinflusst wird. Wird nun bei der Herstellung der Footprints zur Simulation des Prozesses eine signifikante Diskrepanz zwischen Massenflussrichtung und initial definierter z -Achse des Werkzeugkoordinatensystems festgestellt – der Footprint liegt in diesem Fall nicht im „anvisierten“ Punkt auf der Substratoberfläche – so kann diese Diskrepanz durch eine einfache Transformation \mathbf{T}_{WKS} des Werkzeugkoordinatensystems, welche die z -Achse auf die Richtung des Massenflusses ab-

bildet, korrigiert werden. Gegebenenfalls können so für einen Brenner mehrere von Spritzparametern, Düsenaufsätzen etc. abhängige Korrekturen $\mathbf{T}_{\text{WKS}}^i$ erstellt werden. Dieses Vorgehen vereinfacht die Einmessung eines Bauteils sowie die Bahnplanung dahingehend, dass stets eine Ausrichtung des Brenners beziehungsweise der Bahnpunkte normal zur Bauteiloberfläche angestrebt werden kann. Erst durch die (nachträgliche) Wahl der zur Parametrisierung oder beispielsweise zu einem gewählten Düsenaufsatz passenden Korrekturtransformation $\mathbf{T}_{\text{WKS}}^i$ wird die Anstellung des Werkzeugs zur Oberfläche so modifiziert, dass der tatsächliche Spritzstrahl die Bauteiloberfläche senkrecht trifft.

Bei der Festlegung von \mathbf{T}_{WKS} basierend auf den Footprintmessungen ist ferner zu beachten, dass sich im Fall signifikanter Abweichungen zwischen der z-Achse des Werkzeugkoordinatensystems und der Massenflussrichtung durch eine Rotation der Spritzpistole auch eine andere Footprintform ergibt. Gegebenenfalls ist die Herstellung des Footprints also mit der durch \mathbf{T}_{WKS} festgelegten Transformation der Spritzpistole zu wiederholen. Dieser Umstand kann jedoch auch für eine Optimierung der Footprintform genutzt werden. Fasching et al. beschreiben dazu einen Optimierungsansatz, welcher eine optimale Rotation des Brenners berechnet, so dass der resultierende Footprint möglichst symmetrisch ist [54]. Eine Möglichkeit zur Wahl von \mathbf{T}_{WKS} besteht also in der Wahl dieser Rotation verkettet mit einer Translation, welche das Zentrum des resultierenden Footprints in die z-Achse des Werkzeugkoordinatensystems verschiebt.

Insgesamt erlaubt die Bestimmung der Transformation \mathbf{T}_{WKS} zwar eine exaktere Bahnplanung, allerdings ist, um eine wiederholbare Einmessung des Brenners zu gewährleisten, eine Koinzidenz von Massenflussrichtung und Spritzrichtung nicht zwingend erforderlich. Falls sowohl die Footprintversuche als auch die Beschichtung von Bauteilen im produktionstechnischen Einsatz derselben relativen Einmessung des Brenners auf das Bauteil unterliegen, wird auch eine von der Spritzrichtung abweichende Massenflussrichtung simulativ korrekt wiedergegeben. Entscheidend ist daher an dieser Stelle die Definition eines anhand der Brennergeometrie spezifizierten Koordinatensystems, welches im Vorfeld der Beschichtung die Prüfung und Korrektur der Bauteileinmessung erlaubt.

Aufbau der Lasereinmessvorrichtung

Zur Definition eines solchen Koordinatensystems wurde im Rahmen dieser Arbeit die in Abbildung 3.9 dargestellte Einmessvorrichtung entwickelt, welche aus einem zylindrischen Grundkörper besteht, der mittels einer Gewindebuchse an Stelle der Sekundärgasdüse auf die Spritzpistole geschraubt wird. Diese direkte Befestigung der Lasereinmessvorrichtung am Düsengewinde sowie ein zwischen Grundkörper der Laserhalterung und Düsengewinde eingebauter Fixierstift, welcher eine Rotation des Grundkörpers in der Gewindebuchse unterbindet, stellen sicher, dass das durch die Einmessvorrichtung definierte Koordinatensystem relativ zur Spritzpistole mit hoher Wiederholgenauigkeit über mehrere Versuche reproduzierbar ist.

Die Achsrichtungen dieses Koordinatensystems werden durch ein batteriebetriebenes, zylindrisches, zentral im Grundkörper eingebautes Kreuzlasermodul definiert, welches sich zur ein-

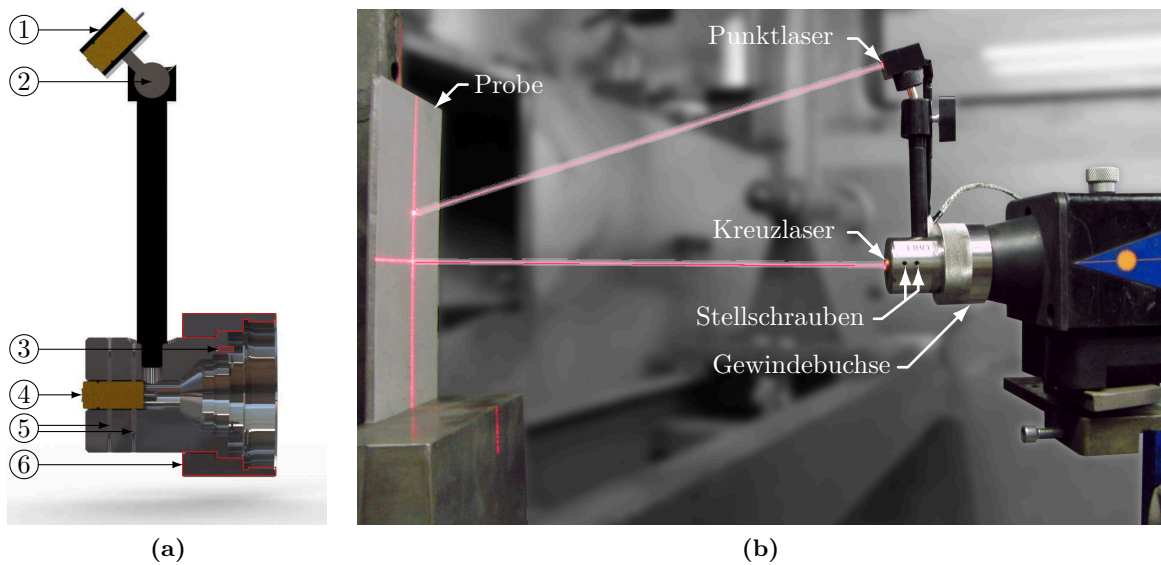


Abbildung 3.9: Entwickelte Lasereinmessvorrichtung. (a): Schnittansicht durch die Einmessvorrichtung mit ① Triangulationslasermodul, ② arretierbarem Kugelgelenk, ③ in das Brennergehäuse greifendem Fixierstift, ④ Kreuzlasermodul, ⑤ insgesamt acht Gewindestiften zur Ausrichtung des Kreuzlasers sowie der ⑥ Gewindebuchse zur Befestigung am Brenner. (b): Auf dem Brenner montierte Vorrichtung bei zu großem Spritzabstand (vergleiche Abbildung 3.10a).

maligen Kalibrierung mit Hilfe von insgesamt acht symmetrisch an vier Seiten des Zylinders eingelassenen Gewindestiften ca. 3° in jede Richtung kippen und zunächst frei drehen lässt. Dieses Kreuzlasermodul strahlt zwei orthogonale Laserebenen in den Raum, deren Schnittgerade die z-Achse des Werkzeugkoordinatensystems definiert und in denen die zur z-Achse orthogonalen x- und y-Achsen des Koordinatensystems liegen. Die Laserebenen werden dabei durch eine Auffächerung eines Laserstrahls durch Brechung an einer Wellenprofilinse erzeugt. Vor der Anfertigung der Footprints wurde dieses Lasermodul einmalig so kalibriert, dass das somit bestimmte Koordinatensystem dem für den Roboter programmierten Werkzeugkoordinatensystem entspricht. Dazu wurde die Pistole waagrecht montiert und das Schlauchpaket so geführt, dass die Pistole möglichst unbeeinflusst bleibt. Durch das robotergesteuerte Hin- und Herdrehen der Pistole um die programmierte Spritzachse beschreibt der näherungsweise senkrecht auf eine entfernte Wand treffende Mittelpunkt des Kreuzlasers nun einen Kreisbogen, dessen Radius um so größer ist, je weiter die durch den Kreuzlaser definierte z-Achse von der programmierten Rotationsachse abweicht. Durch Kippen des Lasermoduls mittels der Gewindeschrauben kann der Laser schließlich parallel zur programmierten Rotationsachse ausgerichtet werden, wobei gleichzeitig seine Rotation um die Spritzachse derart fixiert wird, dass auch die zur z-Achse orthogonalen Achsen des Laserkreuzes den Achsen des Werkzeugkoordinatensystems entsprechen. Dies kann leicht durch das robotergestützte Bewegen des Brenners entlang dieser Achsen überprüft werden. Der genaue Ursprung des nun bis auf eine Translation entlang der z-Achse festgelegten Laserkoordinatensystems, welcher dem TCP

des Werkzeugs entspricht, kann nun durch ein zusätzliches, an einem ca. 10 cm auskragenden Arm schwenkbar befestigtes Punktlasermodul als Schnittpunkt zwischen dem emittierten Laserstrahl und der z-Achse festgelegt werden. Um die genaue Position des Arbeitspunktes auf der z-Achse mit Hilfe des Punktlasers triangulieren zu können, wurde eine Kalibrierspitze angefertigt, welche ebenfalls auf das Düsendengewinde geschraubt werden kann und deren Länge ab der Spitze der Spritzdrähte exakt dem Spritzabstand von 110 mm entspricht. Wird diese Spitze auf einen beliebigen Fixpunkt ausgerichtet, lässt sich nach dem Austausch der Spitze gegen die Lasereinmessvorrichtung die Position des Fixpunktes als TCP übernehmen. Nach dieser Kalibrierung konnte im Bereich des Spritzabstandes keine signifikante mit dem Lasermesssystem messbare Abweichung zwischen den Ausrichtungen des Laserkoordinatensystems und des Werkzeugkoordinatensystems des Roboters mehr festgestellt werden.

Anwendung des Lasereinmesssystems

Das somit eingeführte Koordinatensystem ermöglicht die zuverlässige Kontrolle der relativen Position von Spritzbrenner und Werkstück und damit den Ausgleich potentieller Abweichungen der Brennerausrichtung aufgrund geänderter statischer Belastungen oder aufgrund der nicht formschlüssigen Einspannung des Brenners in der Halterung. Zu diesem Zweck können ein oder mehrere markante Punkte auf der Bauteiloberfläche beziehungsweise -halterung mit dem Roboter angefahren oder die gesamte Roboterbahn langsam nachgefahren werden, um zu überprüfen, ob der von den beiden Lasermodulen projizierte TCP tatsächlich an der gewünschten Bauteilposition liegt. Da die Spritzpistole in der Regel senkrecht zur Bauteiloberfläche angestellt wird, können anhand der Lage des Kreuz- und Punktlasers auf der Oberfläche translatorische und rotatorische Abweichungen jeder Achse des durch die Laser bestimmten Koordinatensystems gegenüber dem programmierten Werkzeugkoordinatensystem des Roboters festgestellt und entsprechend korrigiert werden. Dies wird in Abbildung 3.10 verdeutlicht. Während hierbei Translationen und Rotationen entlang der bzw. um die Spritzachse leicht festzustellen sind (Abbildungen 3.10a und 3.10b), können für die anderen beiden Achsen translatorische von rotatorischen Abweichungen durch Bewegen des Roboters entlang der Spritzachse unterschieden werden. Während bei translatorischen Abweichungen der projizierte Kreuzlasermittelpunkt unverändert bleibt, entfernt er sich bei rotatorischen Abweichungen mit zunehmendem Abstand der Pistole von der Oberfläche weiter von der Sollposition (Abbildungen 3.10c und 3.10d). In der Praxis konnten festgestellte Abweichungen in der Regel auf einfache Weise durch zeitweiliges Lösen der Verschraubung der Halterung und eine Neuausrichtung des Brenners sowie durch die Veränderung der Schlauchpaketführung weitestgehend ausgeglichen werden. Prinzipiell besteht weiterhin die Möglichkeit, die Robotereinmessung oder die Bauteillage zu korrigieren. Insgesamt ermöglicht die entwickelte Lasereinmessvorrichtung somit auf einfache Weise die Erkennung und Kompensation von Abweichungen zwischen Flansch- und Werkzeugkoordinatensystem und stellt die Übertragbarkeit von Footprintexperimenten auf die Beschichtungssimulation und die Bauteilfertigung sicher.

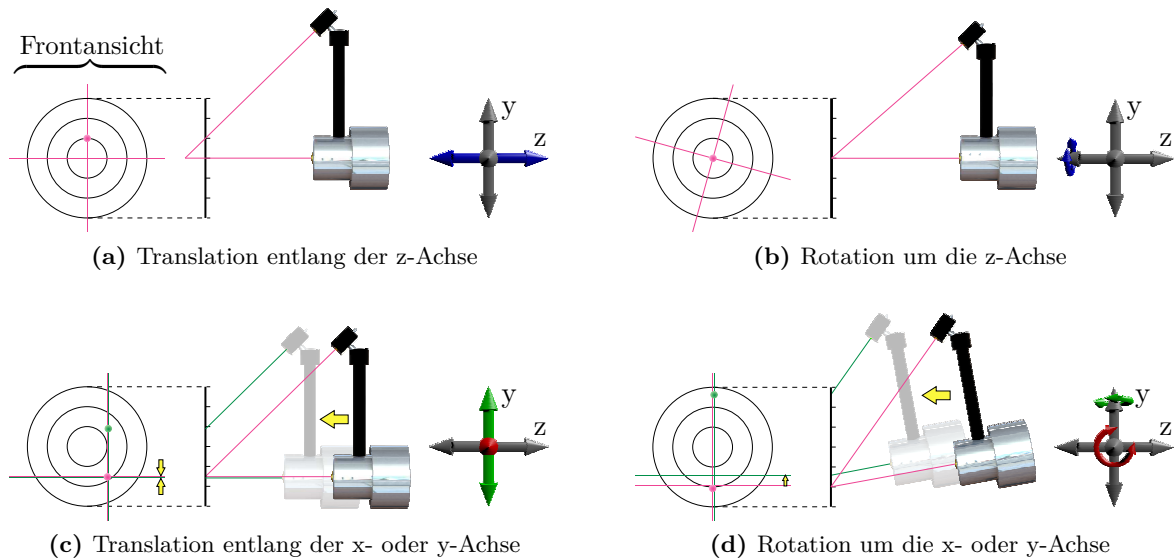


Abbildung 3.10: Detektion und Korrektur von Abweichungen bei der Einmessung eines senkrecht zur Oberfläche angefahrenen Punktes - symbolisiert durch eine Zielscheibe \odot auf der Bauteiloberfläche. Im korrekt eingemessenen Fall treffen sowohl der Kreuz- als auch der obere Triangulationslaser (roter Punkt) den Mittelpunkt. (a) bis (d) zeigen die Positionen des Laserkreuzes und des Triangulationspunktes für alle translatorischen und rotatorischen Abweichungen, von denen jede eindeutig identifiziert werden kann. Während die einfachen Fälle der Translation entlang bzw. Rotation um die Spritzrichtung trivial erkannt und korrigiert werden können ((a) und (b)), müssen seitliche Verschiebungen und Rotationen durch Verfahren des Roboters (gelbe Pfeile) unterschieden werden. Bleibt dabei das Laserkreuz ortsfest auf dem Bauteil, handelt es sich um eine translatorische Abweichung (c), andernfalls um eine rotatorische (d).

3.4.2 Thermische Belastung des Bauteils

Ein weiteres Problem bei der Herstellung eines Footprints stellt die hohe thermische Belastung des Schicht-Substrat-Verbunds dar, welche aufgrund des im Gegensatz zu einer herkömmlichen Bauteilbeschichtung auf einen Punkt konzentrierten Wärmeeintrags besonders groß ist. Daraus ergeben sich zwei unterschiedliche Probleme: Zum Einen kann es im Extremfall zu einem Aufschmelzen und Fließen der Beschichtung kommen und zum Anderen bedingen die durch die Belastung eingebrachten Eigenspannungen eine Verformung des Substrats, wodurch die nachfolgende Vermessung der Footprintprofile deutlich erschwert werden kann. Im Folgenden werden Ursachen und Maßnahmen zur Vermeidung beziehungsweise Verminderung dieser Probleme, welche im Rahmen der Planung des Beschichtungsprozesses zu berücksichtigen sind, beschrieben.

Abbildung 3.11 zeigt zunächst einen Fall, bei dem nach einer Beschichtungsdauer von ca. 14 Sekunden die Oberfläche so heiß ist, dass sie zu schmelzen beginnt. Durch den konzentrierten Gasdruck wird das teigige Material seitlich weggedrückt und es verbleibt ein kleiner Krater in der Oberfläche. Dieser Effekt ist nicht in jedem Fall so leicht erkennbar wie in Abbildung 3.11 und kann sich auch in abgeschwächter Form als leichte Vertiefung in oder nahe der Mitte des

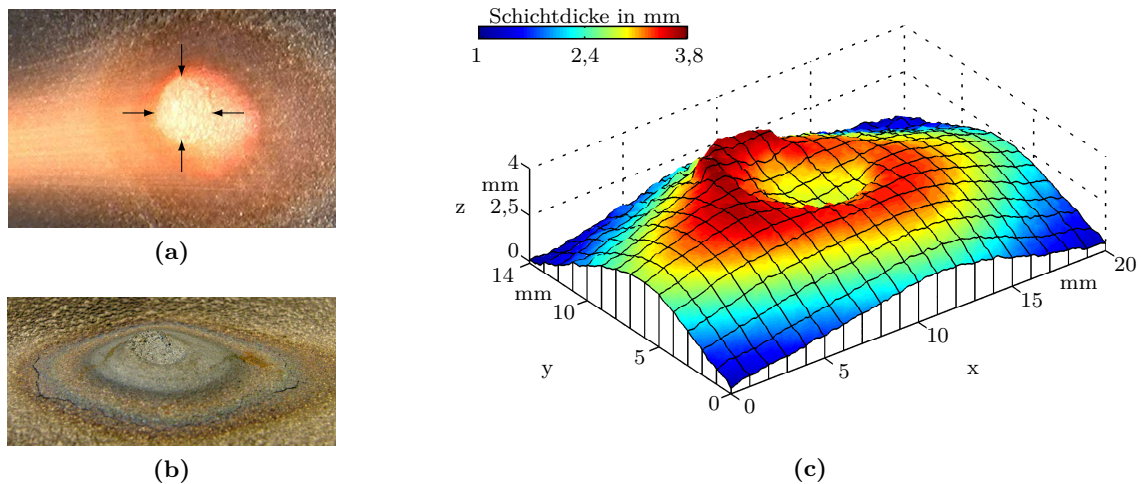


Abbildung 3.11: Aufschmelzung der Beschichtung eines Footprints nach einer Spritzdauer von ca. 14 s. (a) Glühende Probe im Prozess mit geschmolzenem Material (Pfeilmarkierung); (b) Erkalte Probe mit deutlicher Aufwerfung; (c) 3D-Scan der Probe mit ca. 0,8 mm tiefem Krater.

Footprints äußern. Unabhängig von der Stärke der Aufschmelzung ist der Footprint dadurch nicht mehr für die Verwendung in der Beschichtungssimulation geeignet, da sein Profil die Verteilung des Spritzmaterials nicht mehr korrekt wiedergibt.

Um ein solches Aufschmelzen der Oberfläche zu verhindern, bestehen grundsätzlich die Möglichkeiten, die Spritzdauer zu verkürzen oder den Prozess ein- oder mehrmalig zu unterbrechen. Da eine Verkürzung der Spritzdauer auch mit einem dünneren Schichtauftrag einhergeht, welcher bei der späteren Digitalisierung des Schichtprofils einen höheren relativen Messfehler nach sich zieht, ist an dieser Stelle die maximale Spritzdauer zu wählen, bei der eine Verformung der Schichtoberfläche noch ausgeschlossen werden kann. Im Fall der in dieser Arbeit angefertigten Footprints wurden dafür maximal 10 Sekunden als geeignete Spritzzeit ermittelt, welche in einer maximalen Schichtdicke von ca. 3–4 mm für den Lichtbogenspritzprozess resultiert. Eine Unterbrechung des Prozesses, um die Beschichtung zwischenzeitlich abkühlen zu lassen, ist nur empfehlenswert, wenn dabei sichergestellt werden kann, dass die relative Positionierung zwischen Pistole und Substrat dadurch exakt erhalten bleibt – beispielsweise durch das zuvor beschriebene Verfahren, eine Blende im Spritzstrahl zu platzieren. Des Weiteren würde ein Unterbrechen durch Herausbewegen des Substrats aus dem Spritzstrahl und der damit verbundene zusätzliche Auftrag während der Positionierungszeit zu einem höheren Fehler im Materialauftrag führen. Ferner ist zu beachten, dass es zwar prinzipiell möglich ist, mit mehreren Unterbrechungen eine sehr große Schichtdicke zu erzeugen, allerdings kann bei zu großer Schichtdicke nicht mehr von einem näherungsweise homothetischen Wachstum des Profils ausgegangen werden. Dies bedeutet, dass sich das Aufprall- und Haftverhalten der Partikel auf der zunehmend unebenen Schichtoberfläche verändert und die Schichtdicke daher zunehmend nichtlinear mit der Spritzdauer wächst. Derartige, für eine herkömmliche Bauteilbeschichtung atypische Effekte können im Rahmen der Nachbearbeitung der Footprintprofile oder in der Beschichtungssimulation nicht mehr ausgeglichen werden. Vor der Verwendung

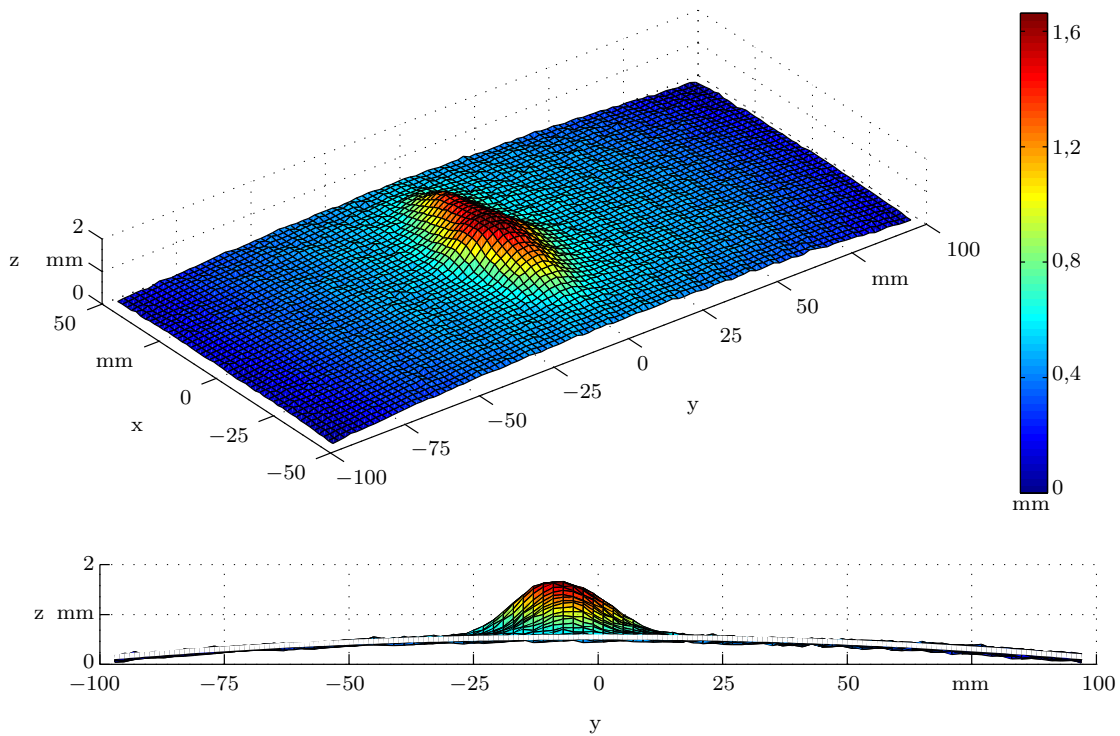


Abbildung 3.12: 3D-Scan eines mit dem Lichtbogenspritzen erzeugten Footprints. Deutlich erkennbar ist die mit ca. 0,54 mm im Vergleich zur maximalen Schichtdicke von 1,15 mm sehr starke Durchbiegung des 5 mm starken Stahlsubstrats aufgrund der hohen thermischen Belastung.

von Footprints in der Simulation sollte daher geprüft werden, ob von einem näherungsweise homothetischen Wachstum der Profile, welches nicht von spezifischen Eingriffsbedingungen der Footprintfertigung dominiert wird, ausgegangen werden kann (vergleiche Kapitel 3.6.4). Im Gegensatz zum Aufschmelzen der Beschichtung sind für die geometrische Verformung des gesamten Schicht-Substrat-Verbunds durch den Prozess eingebrachte Eigenspannungen ursächlich. Für die im Rahmen der Footprintfertigung verwendeten Stahlplatten äußern sich diese in der Regel in Form einer Wölbung der Platten, welche zur Beschichtungsseite oder zur Rückseite weisen kann. Derartige Wölbungen sind – unabhängig von ihrer Richtung – insbesondere im Hinblick auf die Digitalisierung des Footprintprofils mittels Oberflächenprofilometer oder 3D-Scanner problematisch, da nicht mehr von einer ebenen Substratgrundfläche unter dem eingescannten Profil ausgegangen werden kann, sondern sich jede Wölbung des Substrats direkt auf die gemessenen Profildicken auswirkt. Eine konvexe Wölbung des Substrats in Richtung der Beschichtung würde dabei zunächst zu einer Überschätzung und eine konkave Verformung zu einer Unterschätzung der Schichtdicke führen. Im Fall der in dieser Arbeit angefertigten Footprints ergaben sich stets Wölbungen, welche in der Footprintmitte zur Oberseite, also in Richtung der Beschichtung, gerichtet sind (vgl. Kapitel 3.6). Für ein Stahlsubstrat der Größe $150 \times 100 \times 5 \text{ mm}^3$ wurde bei Vorversuchen eine Durchbiegung von

ca. 0,5 mm zwischen der Substratmitte und den Eckpunkten bei einer Spritzdauer von 5 s ermittelt (Abbildung 3.12). Diese Durchbiegung liegt im Bereich der aufgetragenen Schichtdicke und macht eine Vermessung des Schichtprofils mit zerstörungsfreien Methoden nahezu unmöglich.

Die Entstehungsmechanismen dieser Eigenspannungen sind vorwiegend im Rahmen der mit hohen Spannungen einhergehenden Gefahr des Schichtversagens (Delaminierung, Mikrobrüche, Abplatzungen) Gegenstand breiten Forschungsinteresses [29, 62, 86, 95, 99, 183], wobei häufig von der hervorgerufenen Verformung auf den Spannungszustand geschlossen wird. Im Kontext der Herstellung und Vermessung von Footprints ist jedoch primär die Verformung des Werkstücks selbst und nicht der dadurch ersichtliche Spannungszustand von Interesse. Als Entstehungsursachen von Spannungen im Rahmen des thermischen Spritzprozesses sind im Wesentlichen zwei unterschiedliche Effekte bekannt [29, 86]:

1. Splat-interne Zugspannungen

Zugspannungen, welche durch das schlagartige Abkühlen der auftreffenden Partikel in den einzelnen Splats hervorgerufen werden (engl.: quenching stress) und sich in der Schicht akkumulieren.

2. Unterschiedliche Wärmeausdehnungskoeffizienten

Unterschiedliche Wärmeausdehnungskoeffizienten von Beschichtung und Substrat rufen beim gemeinsamen Abkühlen je nach Materialkombination makroskopische Zug- oder Druckspannungen hervor [99].

Abgesehen von diesen während des Spritzprozesses eingebrachten Spannungen ist weiterhin zu berücksichtigen, dass auch durch die Präparation der Probenoberfläche mittels Sandstrahlen oder bereits bei der Fertigung der als Substrat verwendeten Stahlplatten Spannungen induziert werden können. So wurden die im Rahmen dieser Arbeit verwendeten Stahlsubstrate aus kalt gezogenem Stahl hergestellt, welcher produktionsbedingt an der Materialoberfläche unter Zugspannungen steht [167]. Durch das Entfernen des unter Spannung stehenden Materials beim Planfräsen der Werkstückoberfläche kommt es bereits in diesem Schritt zu einer Durchbiegung des Substrates. Ferner stellten Kuroda et al. [87] fest, dass auch durch das Sandstrahlen in einer Tiefe von bis zu 0,5 mm signifikante Druckspannungen eingebracht werden, welche nach Kesler et al. [77] sogar die durch den Spritzprozess eingebrachten Spannungen übertreffen können. Die Überlagerung aller auftretenden Spannungen aus Substratfertigung, Präparation und Spritzprozess ergibt schließlich die resultierende Gesamtverformung und bestimmt auch die Richtung der Wölbung des Substrat-Schicht-Verbunds.

Eine vollständige Vermeidung der Bauteilverformung ist nicht möglich, zur Verminderung der Verformung hat sich jedoch eine Erhöhung der Substratdicke als effektives Mittel erwiesen. Im Vergleich zu der in Abbildung 3.12 gezeigten Probe konnte die Durchbiegung unter Verwendung von Substraten mit einer wesentlich höheren Dicke von 30 mm und einer vergleichbaren Fläche von $150 \times 150 \text{ mm}^2$ trotz der mit 10 s verdoppelten Spritzdauer auf ca. 10–25 μm reduziert werden (vergleiche Kapitel 3.6.2). Die dadurch erreichte Versteifung des Substrats führt

jedoch auch zu größeren Spannungen in der Beschichtung und damit zu vermehrtem Auftreten von Spannungsrissen und einer gesteigerten Gefahr des Abplatzens von Beschichtungsteilen. Im Gegensatz zur Fertigung von Bauteilen im Allgemeinen sind derartige Risse für die Vermessung von Footprints in der Regel unproblematisch, da die Vermessung der Proben keine hohe Belastung darstellt, bei der eine Delamination zu befürchten ist. In seltenen Fällen kam es jedoch bereits während der Abkühlung einer Footprintprobe zu Abplatzungen, welche eine korrekte Vermessung unmöglich machten. Um des Weiteren einer Verformung der Substrate durch im Fräsprozess frei werdende Spannungen entgegenzuwirken, hat es sich bewährt, die Stahlplatten nicht nur auf der Oberseite sondern auch auf der Unterseite mit gleich großer Materialabnahme planzufräsen, um einen symmetrischen Spannungszustand zu erhalten und somit eine Deformation zu minimieren. Es verbleibt jedoch in der Regel eine geringfügige Verformung, für deren Analyse und Korrektur zwei unterschiedliche Ansätze entwickelt wurden, die den Verlauf der Substrat-Beschichtungs-Grenzschicht auf Basis eines ungestrahlten und unbeschichteten Randes des Footprintsubstrats sowie Profilen der unbeschichteten Substrate schätzen. Diese werden in den Kapiteln 3.6.2 und 3.6.3 beschrieben. Zu diesem Zweck wurde jede Probe bereits vor der Beschichtung vermessen und ein 1 cm breiter Rand der Probe, wie in Kapitel 3.3 beschrieben, für den Sandstrahl- und Beschichtungsprozess maskiert.

3.5 Digitalisierung von Footprints

Im Rahmen dieser Arbeit wurden verschiedene optische und taktile Messverfahren und -geräte zur Digitalisierung von Footprintprofilen in Betracht gezogen und bezüglich ihrer Eignung untersucht. Im Folgenden wird zunächst ein Überblick über die unterschiedlichen Systeme gegeben und die ermittelten Vor- und Nachteile dargestellt. Im Anschluss wird auf die Parametrisierung des Scanvorgangs für die gewählten Scansysteme eingegangen.

3.5.1 Auswahl des Scansystems

Zu den getesteten Scansystemen gehören zwei computergesteuerte Konfokalmikroskope auf Basis von Fokusvariation (Alicona⁸ InfiniteFocus und nanoFocus⁹ μ surf), ein Streifenlichtscanner (gom¹⁰ ATOS) sowie zwei taktile Scanner (Renishaw¹¹ Cyclone und Roland¹² Picza PIX-30). Die Eignung der verschiedenen Geräte wurde dabei primär im Hinblick auf die Erfassung dreidimensionaler Footprintprofile der Grundfläche $150 \times 150 \text{ mm}^2$, wie in Abbildung 3.5 dargestellt, untersucht. Tabelle 3.1 zeigt eine Übersicht über die Eckdaten der Geräte und Verfahren.

⁸<http://www.alicon.com>

⁹<http://www.nanofocus.de>

¹⁰<http://www.gom.com>

¹¹<http://www.renishaw.com>

¹²<http://www.rolanddg.com>

3. ERZEUGUNG, DIGITALISIERUNG UND NACHBEARBEITUNG VON FOOTPRINTPROFILIEN

Messgerät	Taktile Scanner		Fokusvariation		Streifenlicht	Querschliff
	Renishaw Cyclone	Roland PIX-30	Alicona InfiniteFocus	nanofocus μ Surf	gom ATOS	Olympus BX51M
Scanvolumen						
lateral (mm ²)	500 × 600	304 × 203	100 × 100	50 × 50	> 10.000	< 40 ^{a)}
vertikal (mm ²)	400	60	3,2 – 22 ^{b)}	75	> 10.000	< 65
Auflösung						
lateral (μ m)	50 ^{d)}	50	0,09 – 3,52 ^{b)c)}	0,2 – 3,1	6 – 28 ^{b)e)}	0,04 – 1,4 ^{b)}
vertikal (μ m)		25 ^{f)}	0,01 – 2,3 ^{b)}	0,001 – 0,02 ^{b)}		
Scandauer ^{g)} (h)	≈ 1	≈ 10	> 20 ^{h)}	> 10 ⁱ⁾	0,5 – 1 ^{k)}	≈ 1 – 2 ^{k)}
Arbeitsmethode	automat.	automat.	automat.	automat.	manuell	manuell

Tabelle 3.1: Vergleich verschiedener Messverfahren und -geräte. ^{a)}Limitiert durch Probeneinbettung; ^{b)}Je nach Objektiv; ^{c)}fixer Abtastpunktabstand; ^{d)}räumliche absolute Genauigkeit, Wiederholgenauigkeit: 5 μ m, Achsgenauigkeit: 1 μ m; ^{e)}räumlicher Antastabstand; ^{f)}fixe Quantisierung; ^{g)}Für einen Footprint 150 × 150 mm²; ^{h)}theoretisch falls kein Systemabsturz; ⁱ⁾mehrere Stunden bei 50 × 50 mm²; ^{k)}manuelle Vermessung / Anwesenheit erforderlich.

Einen Sonderfall stellt in dieser Auflistung die lichtmikroskopische Vermessung des Profils anhand von Querschliffen der Probe dar. Im Gegensatz zu den aufgeführten dreidimensionalen Messverfahren ist diese zweidimensionale Art der Profilmessung in der Literatur im direkten Zusammenhang mit der Simulation des Materialauftrags vorherrschend [46, 47, 54, 158]. Dies hängt damit zusammen, dass viele Schichtauftragssimulationen lediglich zweidimensionale Profile berechnen, um zum Beispiel den Effekt des Bahnabstandes senkrecht zur Simulationsebene verlaufender Bahnen auf die Welligkeit des Gesamtprofils zu beurteilen. Im Hintergrund steht dabei die Beschichtung ebener Flächen auf Basis einer mäanderförmigen Roboterbahn als extrudiertes zweidimensionales Problem. Im Prinzip ist auch eine vollständige dreidimensionale Erfassung einer Probe auf Basis mehrerer Querschliffe durch sukzessives Abschleifen und Aufnehmen dünnster Schichten möglich. So konnte im Rahmen dieser Arbeit beispielsweise mittels Bildmorphingverfahren ein volumetrisches Modell der Schichtstrukturen für einen kleinen Bereich der Beschichtung erstellt werden (Abbildung 3.13) [168]. Die Erfassung vollständiger Footprints ist mit diesem Vorgehen in der Praxis jedoch zu zeitintensiv, da die Probe zwischen je zwei Aufnahmen unter hohem Aufwand weiter abgeschliffen und poliert werden muss. Selbst voll automatisierte Systeme wie z.B. das Robo-MET.3D™-System¹³ benötigen mehrere Stunden pro Millimeter definierten Abtrags durch Schleifen und Polieren [25, 139].

Trotz der im Vergleich zu den übrigen Verfahren fehlenden Dreidimensionalität hat dieses Verfahren jedoch den entscheidenden Vorteil, dass in der Querschliffebene auch Schicht- und Substratstrukturen unterhalb der Oberfläche abgebildet werden. Dies ermöglicht sowohl die Überprüfung der Schichtstruktur auf makroskopische Risse, welche die Messung der abgelagerten Schichtdicke verfälschen könnten, als auch die präzise Messung der im vorherigen Kapitel diskutierten thermisch bedingten Substratverformung. Daher eignen sich Querschliffmessungen gut als Referenzmessungen zur Validierung nicht invasiver Messverfahren – insbesondere

¹³<http://www.ues.com/content/robo-met3d>

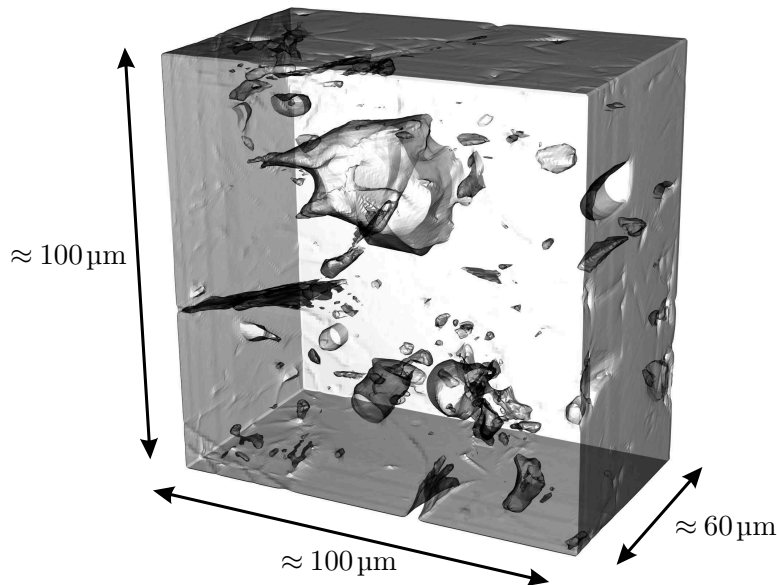


Abbildung 3.13: Rendering des porenhaltigen Modells der ungefähren Größe $100 \times 100 \times 60 \mu\text{m}^3$, welches aus einer Reihe paralleler Querschnittsaufnahmen während des schrittweisen Abschleifens der Schicht erzeugt wurde [168]. Deutlich erkennbar sind unterschiedlich große und unterschiedlich stark verästelte Poren.

im Zusammenhang mit der Kompensation der Substratdurchbiegung – sowie zur Vermessung von Schichtprofilen linearer Bahnverläufe mit annähernd konstantem Querschnittsprofil. Insgesamt kann mit dem eingesetzten Digitalmikroskop BX51M der Firma Olympus¹⁴ mit 5-Megapixel-Kameraaufsatz DP-25 je nach Objektiv ($2,5\times$ bis $100\times$) eine hohe Auflösung von $1,33 \mu\text{m}/\text{Pixel}$ bis $0,033 \mu\text{m}/\text{Pixel}$ erreicht werden. Da zur Vermeidung von Ausbrüchen der Beschichtung durch das Schleifen eine sehr vorsichtige Präparation der Proben erforderlich ist und die Aufnahme eines Querschliffs zur Profilmessung aus einer Menge von Einzelaufnahmen zusammengesetzt werden, muss mit einer relativ langen Bearbeitungszeit von ca. 2 Stunden für die Vermessung eines einzelnen Querschliffs gerechnet werden. Davon entfallen ca. 30–45 Minuten auf die Aufnahme des Querschliffs, während die übrige Zeit auf die Präparation der Probe entfällt. Zusätzlich zur reinen Aufnahmezeit muss das Profil noch vermessen werden. Um diesen Prozess, welcher das Einzeichnen einer Vielzahl von Messpunkten bzw. Abständen entlang des Schichtverlaufs durch den Messtechniker und das manuelle Auswerten dieser Messungen beinhaltet, zu vereinfachen, wurde im Rahmen dieser Arbeit eine Software entwickelt, die diesen Prozess nahezu vollständig automatisiert. Diese wird in Kapitel 3.7 beschrieben. Die beiden in Tabelle 3.1 aufgeführten Konfokalmikroskope der Firmen Alicona und nanofocus arbeiten nach dem Fokusvariationsprinzip [37]. Dabei wird eine Optik eingesetzt, welche einen sehr geringen Schärfentiefenbereich aufweist, so dass nur Objekte in einem eng begrenzten Entfernungsbereich scharf abgebildet werden. Durch die Variation der Entfernung der zu vermessenden Probe zum Objektiv mittels Präzisionsverstelleinheiten wird jeweils nur

¹⁴<http://www.olympus-ims.com/de/microscope/bx61/>

derjenige Teil der Probe scharf abgebildet, welcher sich momentan im Fokus befindet. Auf diese Weise lässt sich – je nach Objektiv – eine hohe Genauigkeit im Submikrometerbereich erzielen. Der Nachteil dieser Geräte besteht allerdings darin, dass sie nur über kleine laterale Messbereiche für Einzelaufnahmen sowie stark beschränkte laterale Fahrwege verfügen, welche die Probengröße deutlich einschränken. Dieser Verfahrensbereich ist bei dem System der Firma Alicona mit $100 \times 100 \text{mm}^2$ zur Aufnahme größerer Footprints theoretisch noch ausreichend, wie sie bei den durchgeführten Experimenten mittels Lichtbogenspritzen vorliegen. Das nanofocus-System eignet sich hingegen mit einem Verfahrensbereich der Größe $50 \times 50 \text{mm}^2$ nicht mehr. In der Praxis treten zudem bei beiden Geräten gravierende Schwierigkeiten beim Scannen großflächigerer Bereiche auf, selbst wenn deren Abmessungen noch wesentlich kleiner als der maximale Verfahrensbereich sind. Die Ursache hierfür liegt darin, dass für eine Probe mit größerer Grundfläche im Scanprozess viele überlappende Einzelaufnahmen zu einem Profil kombiniert werden müssen. In beiden Fällen gelang es den unterschiedlichen Softwaretools der Geräte trotz leistungsfähiger Rechner nicht, diese erforderliche Registrierung der Einzelaufnahmen zu berechnen (Systemabsturz). Beide Geräte sind daher für die Vermessung von Footprintprofilen ungeeignet.

Der Streifenlichtscanner vom ATOS bietet die Möglichkeit, Objekte mit einer Größe von mehreren Metern einzuscannen, und ist diesbezüglich mehr als ausreichend für die Erfassung von Footprints. Zwar nimmt die Genauigkeit mit zunehmender Größe des Messfeldes bzw. der gewählten Objektivs ab, für die Aufnahme der Footprints kann allerdings mit sehr kleinen Messfeldern und einer Genauigkeit im oder nahe am einstelligen Mikrometerbereich gearbeitet werden. Diese vom Hersteller angegebene Genauigkeit gilt jedoch ohne Einschränkungen nur für Einzelbildmessungen. Die Digitalisierung einer Probe erfolgt jedoch durch Aufzeichnen einer Reihe von ca. 10–30 Einzelbildmessungen, für die das System jeweils von Hand neu positioniert und auf einen noch nicht erfassten Teil der Probe ausgerichtet wird. Anhand von auf die Probenhalterung geklebten Referenzpunkten fügt das System die Einzelbilder innerhalb weniger Sekunden automatisch zusammen. Sansoni et al. [124] zeigten, dass bei diesem Prozess die Grundgenauigkeit der zusammengeführten Daten gegenüber den Einzelmessungen wesentlich abnehmen kann – im getesteten Fall etwa um den Faktor 3. Im Rahmen der Digitalisierung von Footprintproben erwies sich ferner die Erfassung der blanken Metalloberfläche des als Referenzfläche unbeschichteten Randes als besonders problematisch. Die hierbei auftretenden, störenden Reflexionen verhindern lokal die Erfassung der Oberfläche und müssen daher durch Aufbringen einer sehr dünnen Schicht eines Antireflexionsmittels unterbunden werden. Dieses Vorgehen ist jedoch nicht perfekt, so dass das erzeugte Dreiecksnetz dessen ungeachtet Löcher an nicht erfassbaren Stellen enthalten kann. Insgesamt ist dieses Verfahren aufgrund der vielfach zu wiederholenden manuellen Neupositionierung, welche nur teilweise den beschriebenen Reflexionen geschuldet ist, relativ zeitintensiv, so dass das Einscannen einer einzelnen Footprintprobe ca. 30–60 Minuten dauert. Aufgrund dieses sehr hohen manuellen Aufwandes beim Scanvorgang ist dieser Scanner im Rahmen dieser Arbeit nicht für die Digitalisierung von Footprintproben verwendet worden. Vergleichbare Geräte, welche eine automatisierte Neupositionierung der Probe sowie kontrollierte Beleuchtungsbedingungen

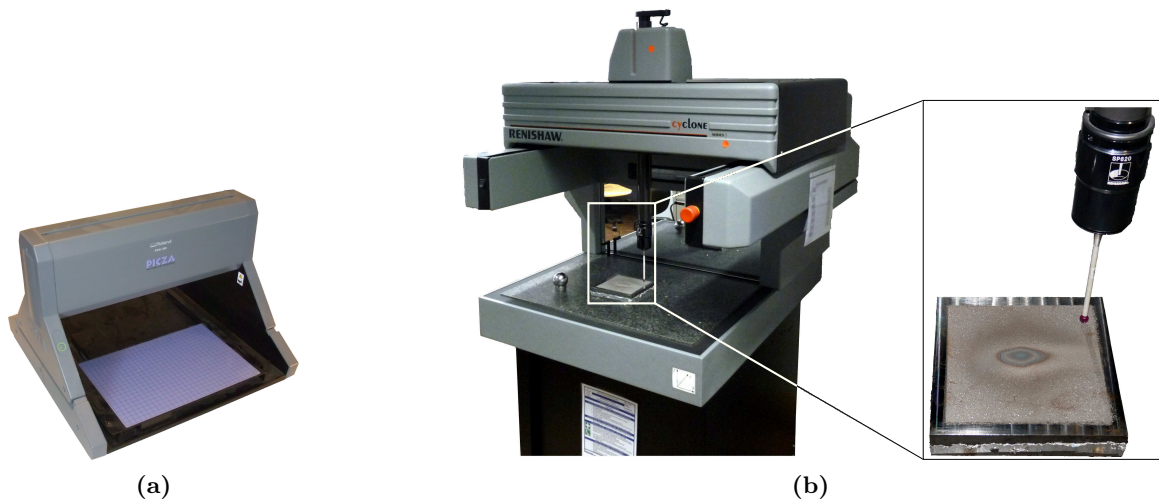


Abbildung 3.14: Für das Scannen von Footprints und anderen Proben verwendete Scansysteme: (a) Nadelscanner PIX-30; (b) Taktile Scanner Renishaw Cyclone mit vergrößert dargestelltem Rubinkugel-Messtaster.

vorweisen, könnten diese Probleme theoretisch beheben.

Mit ca. $20 \times 30 \text{ cm}^2$ und $50 \times 60 \text{ cm}^2$ lateralem Verfahrensbereich weisen auch die beiden taktile Scanner Roland Picza PIX-30 und Renishaw Cyclone einen mehr als ausreichend großen Erfassungsbereich auf (Abbildung 3.14). Während der Picza PIX-30 ein Nadelscanner ist, welcher die Probenhöhe an vordefinierten Punkten eines regulären zweidimensionalen Grundgitters vermisst und daher als Ergebnis ein regelmäßiges Dreiecksnetz ausgibt, handelt es sich bei der Renishaw Cyclone um eine Koordinatenmessmaschine, welche einen Messtaster mit hochgradig abriebfester Rubinkugel an der Spitze entlang einer oder mehrerer definierbarer Richtungen kontinuierlich über die Oberfläche bewegt. Dabei werden innerhalb vorgegebener Toleranzen für den Sehnenfehler automatisch Messpunkte erzeugt. Nach Abschluss des Scanvorgangs wird eine Tasterradiuskompensation durchgeführt, um den Radius der Rubinkugel an der Messtasterspitze auszugleichen. Die Daten werden als Punktwolke oder als trianguliertes Netz ausgegeben. Bei beiden Geräten lassen sich, im Gegensatz zu den Fokusvariatio-
nssystemen und dem Streifenlichtscanner, die laterale und vertikale Auflösung unabhängig voneinander wählen. Dies ist insbesondere für die relativ breiten und flachen Footprintprofile von Vorteil, da durch eine gröbere laterale Auflösung bei gleichbleibend hoher vertikaler Auflösung weniger redundante Datenpunkte anfallen, ohne dass die Messgenauigkeit der Profilhöhe abnimmt. Da der PIX-30 Scanner ein ortsfest quantisiertes, vertikales Scanraster verwendet, sind die ausgegebenen Höhenwerte des resultierenden Netzes stets Vielfache der gewählten Auflösung, welche maximal $25 \mu\text{m}$ beträgt. Demgegenüber unterliegen die Messpunkte der Koordinatenmessmaschine der Firma Renishaw keiner fixen Quantisierung. Die absolute Antastgenauigkeit des Cyclone Systems ist mit $50 \mu\text{m}$ angegeben. Mit einer Antast-Wiederholgenauigkeit von $5 \mu\text{m}$ und einer Achsgenauigkeit von $1 \mu\text{m}$ kann allerdings davon ausgegangen werden, dass eine deutlich höhere relative Genauigkeit erreicht wird, welche im

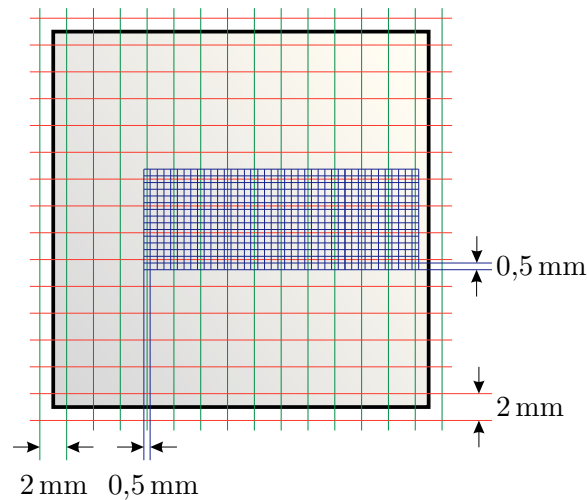


Abbildung 3.15: Schematische, nicht maßstabgetreue Darstellung der vier verwendeten Scanraster für das Scannen eines Footprints. Verwendet wurden zwei orthogonale Raster mit einem Scanbahnabstand von je 2 mm (grün und rot). Zusätzlich wurde der mittlere Bereich auf Basis eines feineren Gitters mit Bahnabstand 0,5 mm hochauflösender abgetastet (blau).

Fall der Footprintprofilmessung von größerer Relevanz ist als die absolute Genauigkeit. Für die Vermessung von Footprints mit einer maximalen Höhe von 1–3 mm ist die Genauigkeit des Cyclone-Systems uneingeschränkt geeignet und die des PIX-30 als grenzwertig aber noch ausreichend zu beurteilen. Des Weiteren arbeiten beide Systeme nach der initialen Programmierung der Scanrichtungen und -bereiche im Gegensatz zum Streifenlichtscanner vollautomatisch, wodurch der Arbeitsaufwand für das Einscannen einer großen Anzahl von Proben deutlich reduziert wird.

Aufgrund der höheren relativen Genauigkeit und der wesentlich kürzeren Scanzeit von ca. einer Stunde wurde daher für alle im Rahmen dieser Arbeit gescannten Footprintprofile das Cyclone System verwendet. Weitere Profilschans, wie die der in Kapitel 3.8 verwendeten Linearbahnen, wurden aus Gründen der besseren Verfügbarkeit mit dem Scansystem Picza PIX-30 durchgeführt. Für die Vermessung der Schichtdickenverteilung auf komplexen Geometrien im Kontext der Evaluation der Beschichtungssimulation wurde ferner ein modernerer Taktile Scanner der Firma Zeiss eingesetzt, welcher in Kapitel 6 beschrieben wird.

3.5.2 Scanprozess

Zur Untersuchung der Planarität der $150 \times 150 \times 30 \text{ mm}^3$ großen Substrate vor dem Spritzprozess sowie zur Ermittlung der im Spritzprozess eintretenden thermischen Verformung wurden die Proben nicht nur nach der Beschichtung, sondern auch im unbeschichteten Zustand eingescannt. Hierfür wurden zwei zueinander orthogonale, parallel zu den Kanten des Substrates verlaufende Scanrichtungen mit einem Scanbahnabstand von 2 mm definiert (Abbildung 3.15). Entlang dieser Scanbahnen wurden Datenpunkte dynamisch in Abhängigkeit des Sehnenfehlers, jedoch mindestens ein Punkt pro Millimeter gesetzt. Die Verwendung von zwei

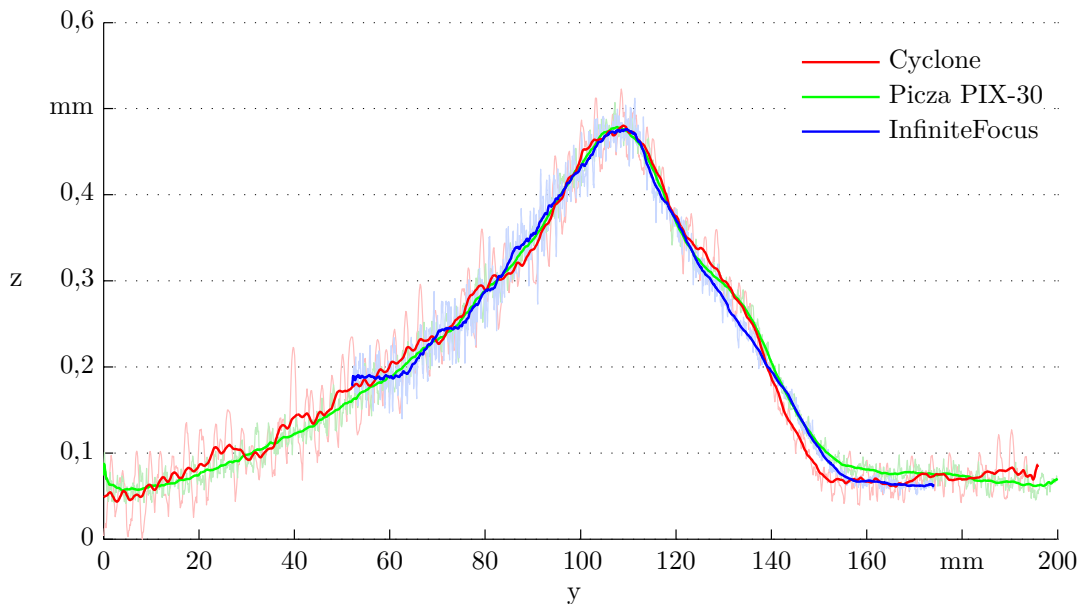


Abbildung 3.16: Querschnittsprofil einer Probe mit linear entlang einer geraden Bahn verfahrenem Spritzbrenner. Die geglätteten Profilmessungen (5 mm breiter gleitender Mittelwertfilter) stimmen im Rahmen der zu erwartenden Messgenauigkeit der verschiedenen Scansysteme gut überein. Zur Verdeutlichung des Rauschens der Oberfläche und der Scansysteme sind die ungefilterten Profildaten im Hintergrund als hellere Kurven dargestellt.

orthogonalen Scanrastern ist hier notwendig, um die Lage aller vier Seitenränder der oberen Beschichtungsfläche der Probe, welche als Referenz für die Ausrichtung der Scandaten dienen, präzise zu erfassen. Mit nur einer Scanrichtung würden die parallel zu dieser Verfahrerrichtung liegenden Kanten aufgrund des 2 mm großen Bahnabstandes nur unzureichend erfasst. Für das Scannen der bereits beschichteten Profile wurde zusätzlich ein noch feineres Abtastgitter für den mittleren Bereich des Footprints mit 0,5 mm Bahn- und Punktabstand erstellt, um die hier vorhandenen feineren Strukturen besser abbilden zu können.

Die Vermessungen der Linearbahnproben in Kapitel 3.8 wurden mit dem PIX-30 Scanner bei maximaler vertikaler Auflösung von 25 μm und lateralem Bahn- und Punktabstand von 0,5 mm auf dem gesamten Scangebiet durchgeführt.

Um sicherzustellen, dass die Messungen der verschiedenen Systeme vergleichbar sind, wurde das Querschnittsprofil einer mit einer einfachen, linearen Beschichtungsbahn erzeugten Probe mit beiden Geräten sowie zusätzlich mit dem wesentlich genaueren InfiniteFocus Profilometer vermessen. Abbildung 3.16 zeigt, dass alle Profile – im Rahmen des Mess- bzw. Oberflächenrauschens – gut übereinstimmen und keine systematischen Abweichungen eines einzelnen Scanners auftreten.

3.6 Nachbearbeitung dreidimensionaler Profile

Nach der messtechnischen Erfassung der Profildaten müssen diese zur Verwendung in der Beschichtungssimulation nachbearbeitet werden. Die hierfür entwickelten Methoden werden im Folgenden näher beschrieben. Zur Nachbearbeitung gehören für die dreidimensionalen Profile zunächst das Ausrichten des Scannerkoordinatensystems am Werkstückkoordinatensystem und die Analyse der unbeschichteten Substrate, welche in den folgenden Kapiteln 3.6.1 und 3.6.2 beschrieben werden. Kapitel 3.6.3 stellt im Anschluss die für die Kompensation thermischer Verformungen entwickelten Methoden vor und Kapitel 3.6.4 beschreibt die Konvertierung der Profildaten für die Nutzung als Footprinttextur im Rahmen der Simulation.

Die experimentelle Grundlage für die Ausführungen in diesen Kapiteln 3.6.1 – 3.6.4 stellt eine Versuchsreihe dar, in der insgesamt zehn Footprintsubstrate vorbereitet und mit variierender Spritzdauer unter Verwendung der in Kapitel 2.2 aufgeführten Standardparametrisierung beschichtet wurden. Die Variation der Spritzdauer diente dabei zum einen der Ermittlung einer geeigneten maximalen Spritzdauer unter Vermeidung des Aufschmelzens und zum anderen der Überprüfung, inwieweit ein näherungsweise homothetisches Schichtwachstum angenommen werden kann. Des Weiteren wurden mehrere der erstellten Proben gemittelt, um den Einfluss von Prozessfluktuationen zu minimieren. Von den zehn gefertigten Proben wiesen zwei Proben Aufschmelzungen auf und konnten daher nicht für die Simulation verwendet werden. Unter den acht geeigneten Footprints sind jeweils drei mit einer Spritzdauer von 5 s und 7,5 s sowie zwei mit einer Spritzdauer von 10 s beschichtet worden.

Nach dem Scanvorgang liegen die Datensätze zunächst als dreidimensionale Punktwolken

$$\mathcal{F}_i = \left\{ \mathbf{x}_j = (x_j, y_j, z_j)^T \in \mathbb{R}^3 \mid j = 1, \dots, n_i \right\}$$

mit jeweils $n_i \approx 90,000$ Punkten vor (Abbildung 3.17). Dabei gibt $i = 1, \dots, 8$ den Index der Footprintprobe an¹⁵. Zusätzlich bezeichnet im Folgenden t_i die Spritzdauer des i -ten Footprints. Initial beschreiben die Punkte \mathbf{x}_j die vom Scanner aufgezeichneten Mittelpunkte der Rubinkugel ($\varnothing 8\text{mm}$) an der Messtasterspitze. Um aus diesen Punkten Oberflächenpunkte der Footprintprobe zu erhalten, muss zunächst eine Tasterradiuskompensation durchgeführt werden. Hierfür wurde eine im Rahmen des Sonderforschungsbereichs 708 am Institut für Spanende Fertigung¹⁶ entwickelte Software eingesetzt, welche in jedem Punkt \mathbf{x}_j einen Normaleneinheitsvektor \mathbf{n}_j auf dem durch die Punktwolke gegebenen lokalen Flächenverlauf approximiert [60] und anschließend alle Punkte entlang dieser Normalen um den Tasterradius von 4 mm verschiebt: $\mathbf{x}_j \mapsto \mathbf{x}_j + 4\mathbf{n}_j$. Für die mit dem Nadelscanner PIX-30 erfassten Profile ist dieser Schritt, aufgrund des sehr geringen Durchmessers der Messnadel, nicht erforderlich.

¹⁵In den folgenden Abschnitten werden die Footprints bzw. die sie beschreibenden Punktmengen schrittweise Transformationen unterzogen, wobei sich sowohl die Anzahl der in den Mengen \mathcal{F}_i enthaltenen Punkte n_i als auch die Punkte \mathbf{x}_j selbst ändern können. Aus Gründen der Einfachheit werden jedoch die Notationen \mathcal{F}_i , n_i und \mathbf{x}_j beibehalten und beziehen sich stets auf die „aktuellen“ aus dem Kontext hervorgehenden Datensätze.

¹⁶<http://www.isf.de>

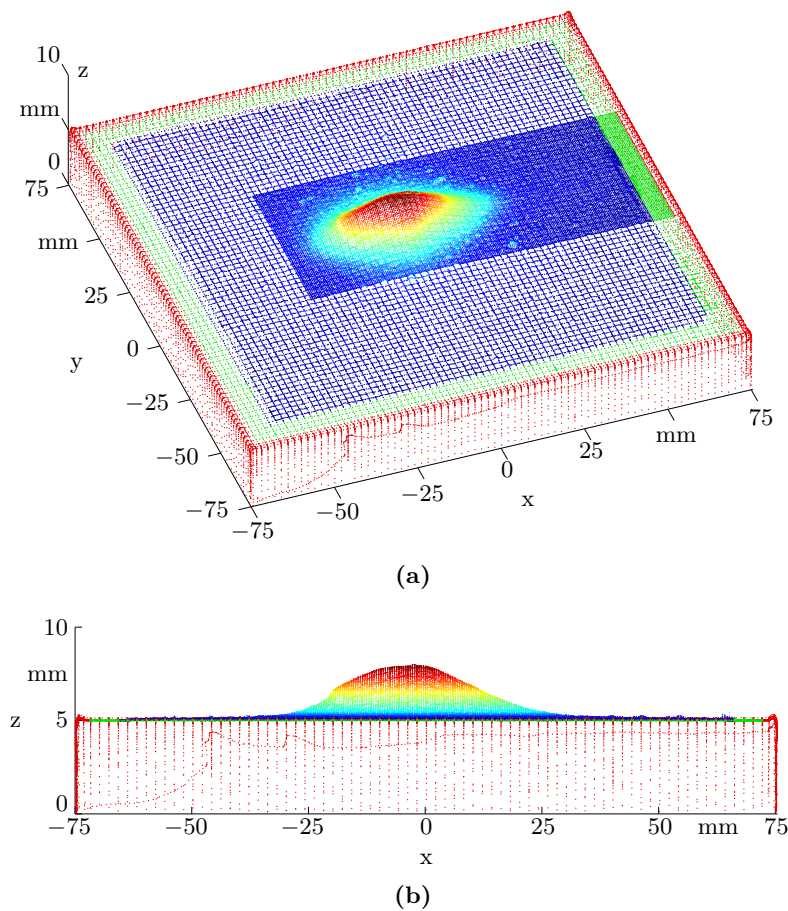


Abbildung 3.17: (a) Schrägansicht und (b) Seitenansicht des dreidimensionalen Scans eines Footprints mit 7,5 s Spritzdauer als Punktwolke bestehend aus 94.607 Abtastpunkten. Die nach der Ausrichtung des Datensatzes verworfenen Seitenflächen sind rot und die unbeschichtete Referenzfläche R_i ist grün hervorgehoben. Der mittlere blaue Bereich stellt die ca. $130 \times 130 \text{ mm}^2$ große Nutzfläche der Probe dar. Deutlich zu erkennen ist das mit 0,5 mm statt 2 mm Scanlinienabstand feinere Abtastgitter im mittleren Bereich des Footprints.

3.6.1 Ausrichtung der digitalisierten Profile

Da die im Scanner platzierten Proben mit ihren Kanten nicht absolut präzise an den Koordinatenachsen des Scanners ausgerichtet werden können und zudem Unebenheiten – beispielsweise durch Spritzmaterialanhaftungen oder minimale Kleberrückstände des Maskierungsbandes auf der Unterseite der Proben – zu einer leichten Neigung der Probe führen können, besteht der erste Schritt darin, die digitalisierten Profile achsparallel auszurichten. Nach der händischen Entfernung offensichtlicher Ausreißer werden die im Scannerkoordinatensystem vorliegenden Daten \mathcal{F}_i dabei in zwei Schritten in das Werkstückkoordinatensystem überführt, bei welchem die x- und y-Achse parallel zu den Seitenflächen des Footprints verlaufen und die Schichtdicke entlang der z-Achse gemessen werden kann (Abbildung 3.17). Zur Ausrichtung entlang der

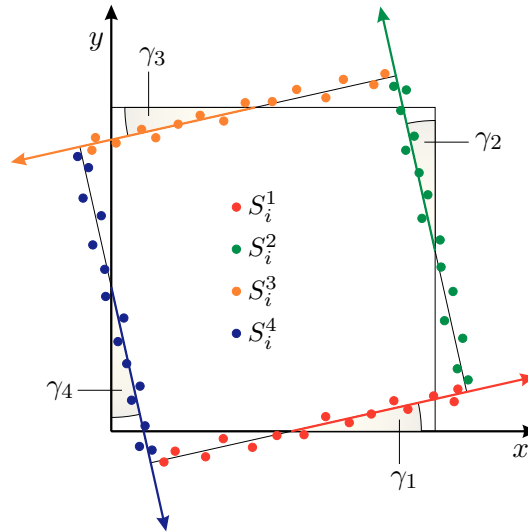


Abbildung 3.18: Ausrichtung der x- und y-Achsen des Scannerkoordinatensystems. Der Rotationswinkel γ um die z-Achse (senkrecht zur Bildebene) wird als Mittelwert der Winkel $\gamma_1, \dots, \gamma_4$ zwischen den ersten Hauptkomponentenrichtungen (farbige Vektoren) der Seitenflächen S_i^1, \dots, S_i^4 und den x- Achsen bzw. y-Achsen bestimmt.

x- und y-Achsen wird die Punktwolke zunächst um die z-Achse rotiert. Der hierfür benötigte Rotationswinkel wird als Mittelwert der vier Winkel zwischen den Seitenflächen und den jeweils nächstgelegenen x- oder y-Achsen bestimmt (Abbildung 3.18). Zu diesem Zweck werden Teilmengen S_i^1, \dots, S_i^4 der Punkte aus \mathcal{F}_i bestimmt, welche die vier äußeren Seitenflächen S_i^1, \dots, S_i^4 definieren. Die Punkte jeder Seitenfläche werden anschließend in die x, y -Ebene projiziert und ihre Ausrichtung innerhalb dieser Ebene wird mittels Hauptkomponentenanalyse bestimmt. Aus den Winkeln $\gamma_1, \dots, \gamma_4$ zwischen den ersten Hauptkomponenten und den Koordinatenachsen wird schließlich ein mittlerer Rotationswinkel γ bestimmt, welcher im Fall der acht betrachteten Footprints im Bereich von 0° – 2° lag.

Um die Probe des Weiteren so auszurichten, dass entlang der z-Achse die Profilhöhe über dem unbeschichteten Rand der Probe gemessen werden kann, werden die Hauptkomponenten $\mathbf{p}_1, \dots, \mathbf{p}_3$ aller Punkte des unbeschichteten Randes R_i (Abbildung 3.17), welcher als Referenzfläche dient, in der Reihenfolge absteigender Varianz ermittelt. Die ersten beiden Vektoren \mathbf{p}_1 und \mathbf{p}_2 spannen eine Ausgleichsebene für die Randpunkte und damit die Substratoberfläche auf. Die dritte, zu dieser Ebene senkrecht stehende Hauptkomponente \mathbf{p}_3 minimaler Varianz gibt die Messrichtung für die Schichtdicke an. Daher wird der Datensatz so rotiert, dass \mathbf{p}_3 anschließend parallel zur z-Achse verläuft. Als Rotationsachse dient die in der Ausgleichsebene durch den Mittelpunkt über der Substratfläche verlaufende Gerade mit der Richtung $\mathbf{p}_3 \times (0, 0, 1)^T$. Der Rotationswinkel beträgt

$$\alpha = \cos^{-1} \left(\left\langle (0, 0, 1)^T, \mathbf{p}_3 \right\rangle \right).$$

Für die acht gemessenen Datensätze ergaben sich hierbei Rotationen um im Durchschnitt $\alpha = 0,0178^\circ$. Die maximal auftretende Rotation betrug $\alpha_{\max} = 0,036^\circ$. Obwohl diese Rotatio-

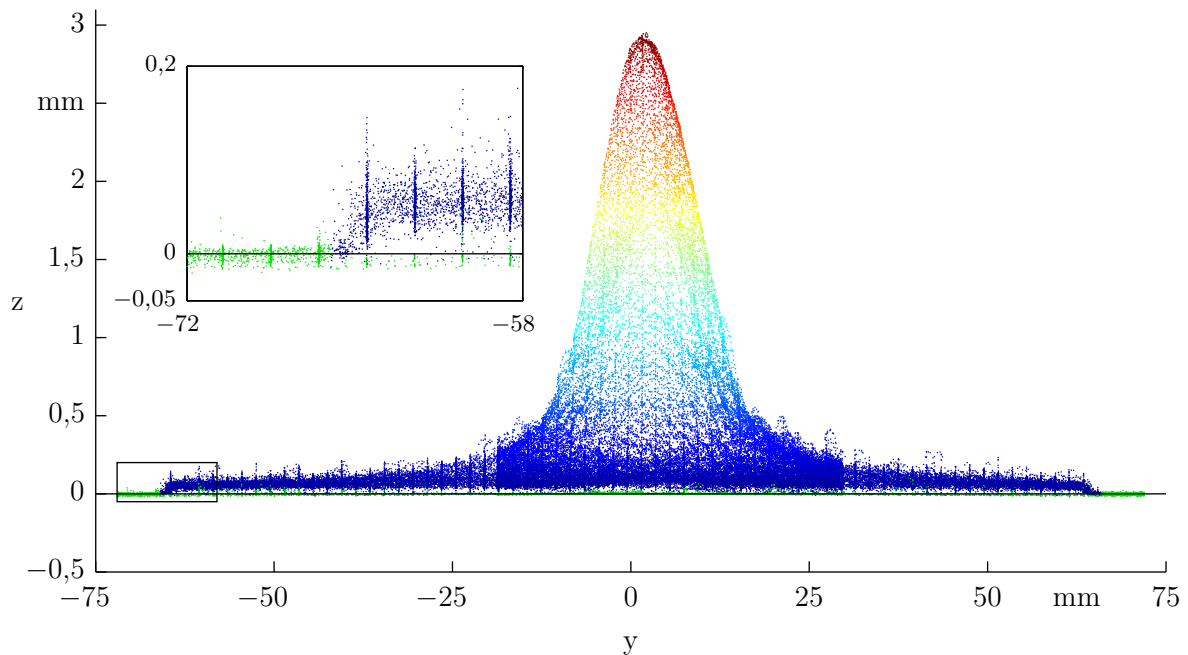


Abbildung 3.19: Scandatensatz des Footprints aus Abbildung 3.17 in der Seitenansicht. In der Vergrößerung ist zu erkennen, dass die grünen Punkte des unbeschichteten Referenzrahmens erfolgreich an der Ebene $z = 0$ ausgerichtet wurden. Der in der Vergrößerung des Randgebietes ersichtliche sprunghafte Anstieg der gemessenen Profilhöhe auf ca. $75\ \mu\text{m}$ stellt keinen Anstieg in der Schichtdicke dar, sondern wird durch das Sandstrahlen der Probe verursacht. In die stark aufgeraute Oberfläche kann die Tastspitze des Scanners mit einem Durchmesser von $8\ \text{mm}$ nicht mehr eindringen. Die erkennbaren senkrechten „Streifen“ sind Häufungen der Scanpunkte entlang der senkrecht zur Abbildungsfläche verlaufenden Scanrasterlinien im Abstand von $2\ \text{mm}$.

nen zunächst nur sehr geringfügig erscheinen, haben sie große Auswirkungen auf die Profilmessung. Für die in der Simulation nutzbare Profilfläche von ca. $130 \times 130\ \text{mm}^2$ ändert sich die Differenz der Profilhöhe beziehungsweise der z -Koordinaten diagonal gegenüberliegender Eckpunkte durch die Rotation im Durchschnitt um $57\ \mu\text{m}$ und für α_{\max} um ca. $116\ \mu\text{m}$. Im Vergleich zur maximalen Footprinthöhe von ca. $400\ \mu\text{m}$ pro Sekunde Spritzdauer sowie zu typischen Schichtdicken von $400\text{--}600\ \mu\text{m}$ zeigt sich jedoch die Wichtigkeit dieses Korrekturschritts.

Nach der Rotation des Datensatzes wird dieser erneut in x - und y -Richtung zentriert. Anschließend werden die in Abbildung 3.17 rot dargestellten Punkte der Seitenflächen verworfen und die Daten in z -Richtung verschoben, so dass die Punkte des Randes R_i im Mittel bei $z = 0$ liegen. Abbildung 3.19 zeigt einen der ausgerichteten Datensätze.

3.6.2 Analyse der unbeschichteten Substrate

Die unbeschichteten und ungestrahlten Substrate unterliegen, wie in Kapitel 3.4.2 erläutert, bereits Spannungen, welche aus dem Fertigungsprozess des kaltgezogenen Stahls resultieren und durch das Abnehmen von Material bei der Fräsbearbeitung der Substrate frei werden.

Weiterhin hat die Aufrauung der Substratoberfläche vor dem Beschichten mittels Sandstrahlen einen Einfluss auf die Geometrie der Probenoberfläche und insbesondere die Scanabtastung der Proben. Um die Einflüsse dieser vorverarbeitenden Prozesse auf die Substratform und das Scanbild der beschichteten Proben einschätzen und kompensieren zu können, wurden zum einen alle Substrate bereits vor dem Sandstrahlen vermessen und zum anderen wurde anhand einer separaten Probe gleichen Materials der Effekt des Sandstrahlens untersucht.

Einfluss des Sandstrahlens

Das Sandstrahlen der Testprobe fand unter den gleichen Bedingungen wie das Strahlen der Footprints substrate mit den in Kapitel 2.2 aufgeführten Parametern statt. Um den Unterschied zwischen der gefrästen Oberfläche und der gestrahlten Oberfläche zu analysieren, wurde ein Teil der Referenzprobe vor dem Strahlen mit einem Stahlblech abgedeckt und anschließend ein ca. $4 \times 16 \text{ mm}^2$ großer Bereich an der Grenze zwischen den aufgerauten und blanken Zonen mit dem Alicona InfiniteFocus Konfokalprofilometer eingescannt. Hierbei wurde eine hohe vertikale Auflösung von ca. $2,57 \mu\text{m}$ erreicht. In dem in Abbildung 3.20a dargestellten Höhenbild des Scans befindet sich auf der rechten Seite der ungestrahlte Bereich der Probe, in dem lediglich leichte Fräsriefen des Fräsprozesses sichtbar sind. Im linken, gestrahlten Bereich hingegen ist die Aufrauung der Oberfläche deutlich zu erkennen, wobei durch das Auftreffen des Strahlguts Kavitäten und Aufwerfungen entstehen, die bis zu $100 \mu\text{m}$ von der ursprünglichen mittleren Höhe abweichen. Abbildung 3.20b zeigt das dreidimensionale Profil in der Seitenansicht. Ebenfalls eingezeichnet ist die maßstabsgetreu abgebildete Rubintasterkugel ($\varnothing 8 \text{ mm}$) des taktilen Scanners. Dabei wird deutlich, dass der Messtaster die hochfrequenten Anteile der Oberfläche nicht erfassen kann, sondern auf den durch das Strahlen gebildeten Spitzen entlangläuft. Eine geometrische Simulation dieses taktilen Abtastvorgangs auf Basis der hochaufgelösten Scandaten des Konfokalprofilometers im Bereich des gestrahlten Teils der Oberfläche ergab, dass hieraus eine Überschätzung der Profilhöhe von durchschnittlich $\gamma = 76,6 \mu\text{m}$ gegenüber der ursprünglichen Höhe der ungestrahlten Oberfläche resultiert. Dieser Effekt zeigt sich bei allen gescannten Footprints an den Übergängen zwischen den unbeschichteten Probenrändern und dem gestrahlten Bereich. Obwohl der äußere Bereich der Proben unbeschichtet ist, wird im Scanbild ein sprunghafter Anstieg des Profils um ungefähr γ verzeichnet (Vergrößerung in Abbildung 3.19). Für die Profile dieser Footprints kann daher näherungsweise angenommen werden, dass Bereiche mit einer Höhe $z < \gamma$ über dem in den Randpunkten zentrierten Nullpunkt unbeschichtet sind. Die Profilhöhen dieser Punkte können aus diesem Grund prinzipiell auf Null gesetzt werden. Die genaue Vorgehensweise zur Ermittlung *unbeschichteter* Bereiche wird für die aus mehreren Einzelprofilen gemittelten Footprints \bar{F} allerdings erst nach der Korrektur der thermischen Verformung durchgeführt und in Kapitel 3.6.4 beschrieben. Für die übrigen, *beschichteten* Bereiche der Probe wird davon ausgegangen, dass die Beschichtung die durch das Sandstrahlen entstandenen Kavitäten weitestgehend ausfüllt und so eine formschlüssige, verklammernde Verbindung zwischen Beschichtung und Substrat herstellt. Dies bedeutet insbesondere, dass der Grenzverlauf zwischen Substrat und Beschichtung im

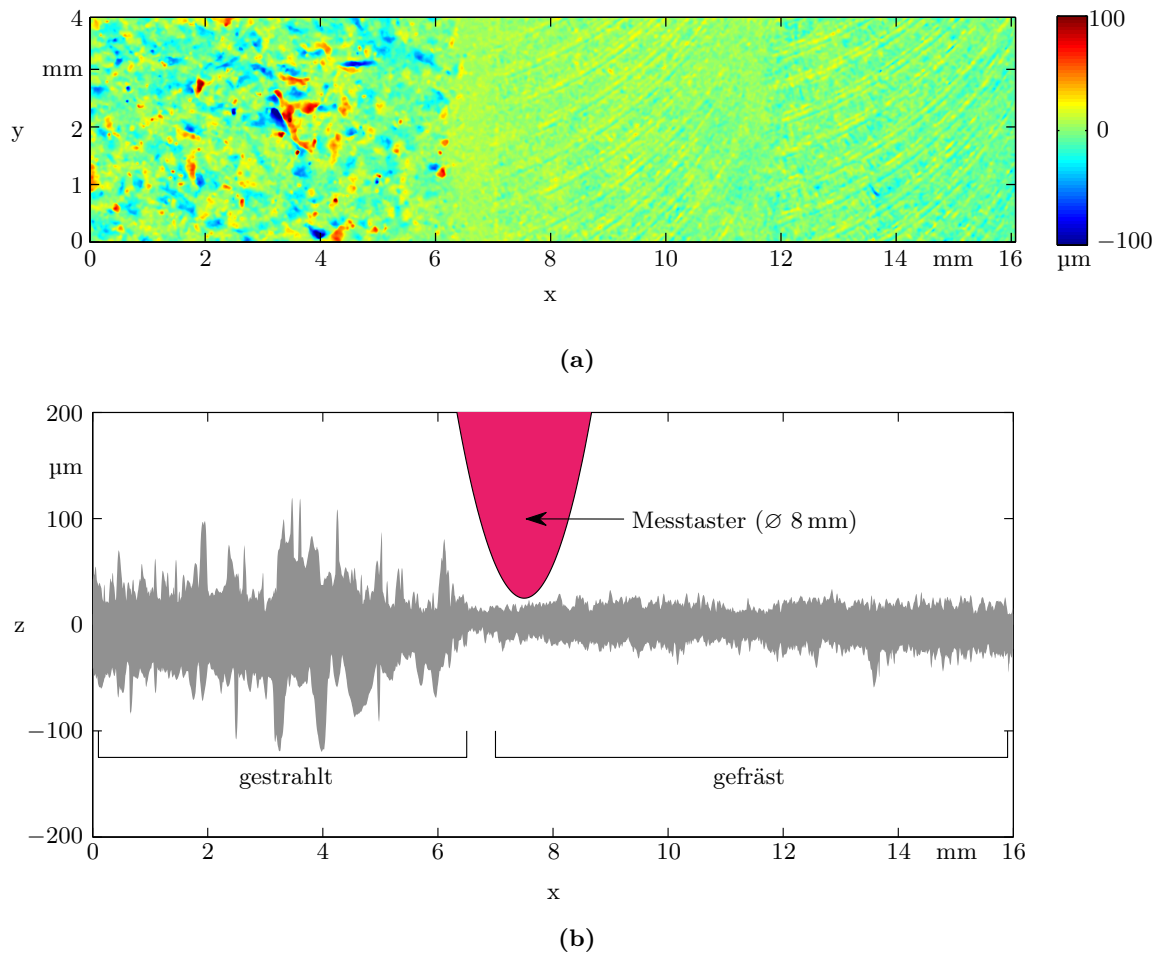


Abbildung 3.20: Profilbilder einer Probe mit sandgestrahltem und ungestrahltem, plangefrästem Bereich. (a) Die gesteigerte Rauheit mit Spitzen und Tälern bis zu $\pm 100 \mu\text{m}$ sowie flache kreisbogenförmige Fräsriefen sind erkennbar. (b) Seitenansicht mit maßstabsgetreu abgebildeter Rubintasterkugel mit Durchmesser 8 mm.

Mittel dem Verlauf der Substratoberfläche vor dem Sandstrahlen entspricht und an diesen Stellen keine Korrektur der Profilhöhe notwendig ist. Unberücksichtigt bleiben jedoch eventuelle Abtastungenauigkeiten auf der Oberfläche der Beschichtung, da die hier vorhandene Rauheit nicht homogen ist, sondern unter anderem von der Schichtdicke abhängt. Oberflächenrauheiten für beschichtete komplexe Bauteile werden zwar in Kapitel 6.3.2 untersucht, eine Übertragbarkeit auf Footprintversuche ist jedoch aufgrund der unterschiedlichen Eingriffsbedingungen nicht gegeben.

Verformungsanalyse der unbeschichteten Substrate

Neben der Untersuchung der durch das Sandstrahlen verursachten Messabweichungen wurden für die Analyse und den Ausgleich der thermischen Verformungen die Profile der unbeschichteten Substrate herangezogen, welche nach der Fräsbearbeitung aber noch vor dem

Fehlermaß	Polynom 2. Grades (k=2)		Polynom 3. Grades (k=3)	
	Alle Punkte	Randpunkte	Alle Punkte	Randpunkte
E_{SSD} / μm^2	52,45	118,71	46,94	114,66
E_{AAD} / μm	1,43	2,30	1,34	2,26
z_o / μm	3,54	4,93	3,31	4,79

Tabelle 3.2: Approximationsfehler der Substratoberfläche polynomieller Approximationsfunktionen zweiten und dritten Grades, welche auf Basis aller Scanpunkte oder nur auf Basis der Randpunkte berechnet wurden.

Sandstrahlen ermittelt wurden. Dabei kam derselbe Scanner wie bei der Digitalisierung der fertiggestellten Footprints mit identischen Parameterwerten zum Einsatz, wobei auf das dritte, feinere Scanraster mit 0,5 mm Bahnabstand aufgrund der fehlenden stärkeren Krümmungen des Footprints verzichtet wurde. Bei der Auswertung der Scans des unbeschichteten Substrates zeigte sich, dass alle Substrate nach der Fräsbearbeitung bereits eine leichte Durchbiegung aufwiesen. Für eine Charakterisierung und Quantifizierung dieser Durchbiegung wurden die verrauschten Punktwolkendatensätze G_i der unbeschichteten Substrate zunächst durch Polynomfunktionen $g(x, y)$ über der x, y -Ebene approximiert. Hierfür wurden zunächst Polynome zweiten und dritten Grades, also der Form

$$g^2(x, y) = a_{20}x^2 + a_{02}y^2 + a_{11}xy + a_{10}x + a_{01}y + a_{00} \quad \text{bzw.}$$

$$g^3(x, y) = a_{30}x^3 + a_{03}y^3 + a_{21}x^2y + a_{12}xy^2 + a_{20}x^2 + a_{02}y^2 + a_{11}xy + a_{10}x + a_{01}y + a_{00}$$

in Betracht gezogen. Im Hinblick auf die Schätzung des genauen Substratverlaufs „unter“ der Beschichtung für die Footprintscans \mathcal{F}_i in Kapitel 3.6.3 stellt sich die Frage, wie gut ein Fittingpolynom, welches nur auf Basis der Punkte eines ca. 1 cm breiten Randes der Oberfläche berechnet wurde, den mittleren Bereich der Oberfläche approximiert. Insbesondere ist dabei der Unterschied zur Approximationsgüte von Polynomfittings von Interesse, welche auf Basis aller Scanpunkte berechnet wurden. Tabelle 3.2 enthält hierzu eine Übersicht über die berechneten Approximationsfehler für Polynome zweiten und dritten Grades, welche jeweils auf Basis aller Punkte oder nur auf Basis der Randpunkte berechnet wurden. Diese auf den *unbeschichteten* Proben berechneten Daten geben einen Hinweis auf die erreichbare Approximationsgüte einer randbasierten Schätzung für die *beschichteten* Proben, für die diese Abschätzung aufgrund der bereits aufgetragenen Beschichtung nicht berechnet werden kann. Als globale Fehlermaße dienen hier die arithmetischen Mittel der Fehlermaße aller acht Einzelfootprints, wobei folgende Fehlermaße berechnet wurden:

- Die Summe quadratischer Abweichungen der Scanpunkte des i -ten Footprints vom zugehörigen Fittingpolynom g_i^k k -ten Grades

$$E_{SSD}^i(k) = \frac{1}{n_i} \sum_{\mathbf{x}_j \in \mathcal{F}_i} \left(g_i^k(x_j, y_j) - z_j \right)^2.$$

- Die mittlere absolute Abweichung aller Scanpunkte

$$E_{AAD}^i(k) = \frac{1}{n_i} \sum_{\mathbf{x}_j \in \mathcal{F}_i} \left| g_i^k(x_j, y_j) - z_j \right|.$$

- Die Obergrenze $z_o^i(k)$ eines 95 %-Konfidenzintervalls für $E_{\text{AAD}}^i(k)$, so dass also 95 % aller Scanpunkte in z -Richtung näher als $z_o^i(k)$ an $g_i^k(x, y)$ liegen.

Wie zu erwarten ist, weisen die Fittings basierend auf allen Scanpunkten gegenüber den Fittings auf den Randpunkten R_i einen geringeren Fehler auf. Quantitativ ist der durchschnittliche absolute Fehler bei Verwendung der Randpunkte ca. 60–70 % größer. Deutlich weniger ausgeprägt ist der Vorteil von $g^3(x, y)$ gegenüber $g^2(x, y)$ – im wichtigen Fall der randbasierten Approximation ist der mittlere absolute Fehler gerade einmal 1,8 % kleiner. Aufgrund dieses geringen Vorteils sowie der Tendenz der berechneten Approximationen höheren Grades $k = 3, \dots, 5$ zu physikalisch unbegründeten Schwankungen mit teils mehreren Extremstellen im mittleren Bereich der Substratfläche werden für das weitere Vorgehen stets Approximationen zweiten Grades verwendet.

Anhand der berechneten Polynomapproximationen, welche beispielhaft für drei Substrate in Abbildung 3.21 dargestellt sind, ist erkennbar, dass die bereits vor der Beschichtung vorhandene leichte Verbiegung der Substrate unterschiedliche Formcharakteristiken aufweist. Sie kann elliptisch, sattelförmig oder auch nahezu zylindrisch sein, verschiedene Ausrichtungen einnehmen und konvex oder konkav sein (Abbildung 3.21a). Letzteres hängt dabei nicht vom Material oder Fräsprozess ab, sondern von der zufälligen Wahl der zu beschichtenden Seite des Substrates. Fasst man für eine quantitative Betrachtung die *Durchbiegung* als Breite des Intervalls aller Höhenwerte einer Probe auf, so liegt diese bei allen Proben zwischen 10,4 μm und 23,6 μm . Im Mittel aller Substrate beträgt die Durchbiegung 18,1 μm . Bei den Randpunkt-basierten Fittings (Abbildung 3.21b) stimmen die Ränder weitestgehend mit denen aus Abbildung 3.21a überein. In der Mitte ergeben sich jedoch erwartungsgemäß leichte Abweichungen, welche unter Umständen in einer anderen Formcharakteristik resultieren. Unter Berücksichtigung der im Vergleich zur Footprintfläche sehr geringen vertikalen Durchbiegung handelt es sich hierbei jedoch nicht um gravierende Approximationsfehler sondern um ein zufälliges „Umklappen“ der Charakteristik. Die Ausrichtung der Charakteristik bleibt in allen Fällen erhalten.

3.6.3 Korrektur thermischer Verformungen

Die durch das thermische Spritzen eingebrachten Verformungen des Substrates führen, wie bereits in Kapitel 3.4.2 diskutiert worden ist, dazu, dass die Schichtdicke nicht mehr direkt aus der Profilhöhe der durchgeführten Scans bestimmt werden kann. Dies ist darin begründet, dass sich das Substrat „unter“ der Beschichtung nach oben bzw. nach unten gewölbt haben könnte, wodurch die Schichtdicke über- respektive unterschätzt wird. Zur Berücksichtigung dieses Effektes wird daher der tatsächliche Verlauf der Substrat-Schicht-Grenze geschätzt und von der gemessenen Profilhöhe abgezogen. Hierzu wurden zwei Methoden entwickelt, die im Folgenden beschrieben werden.

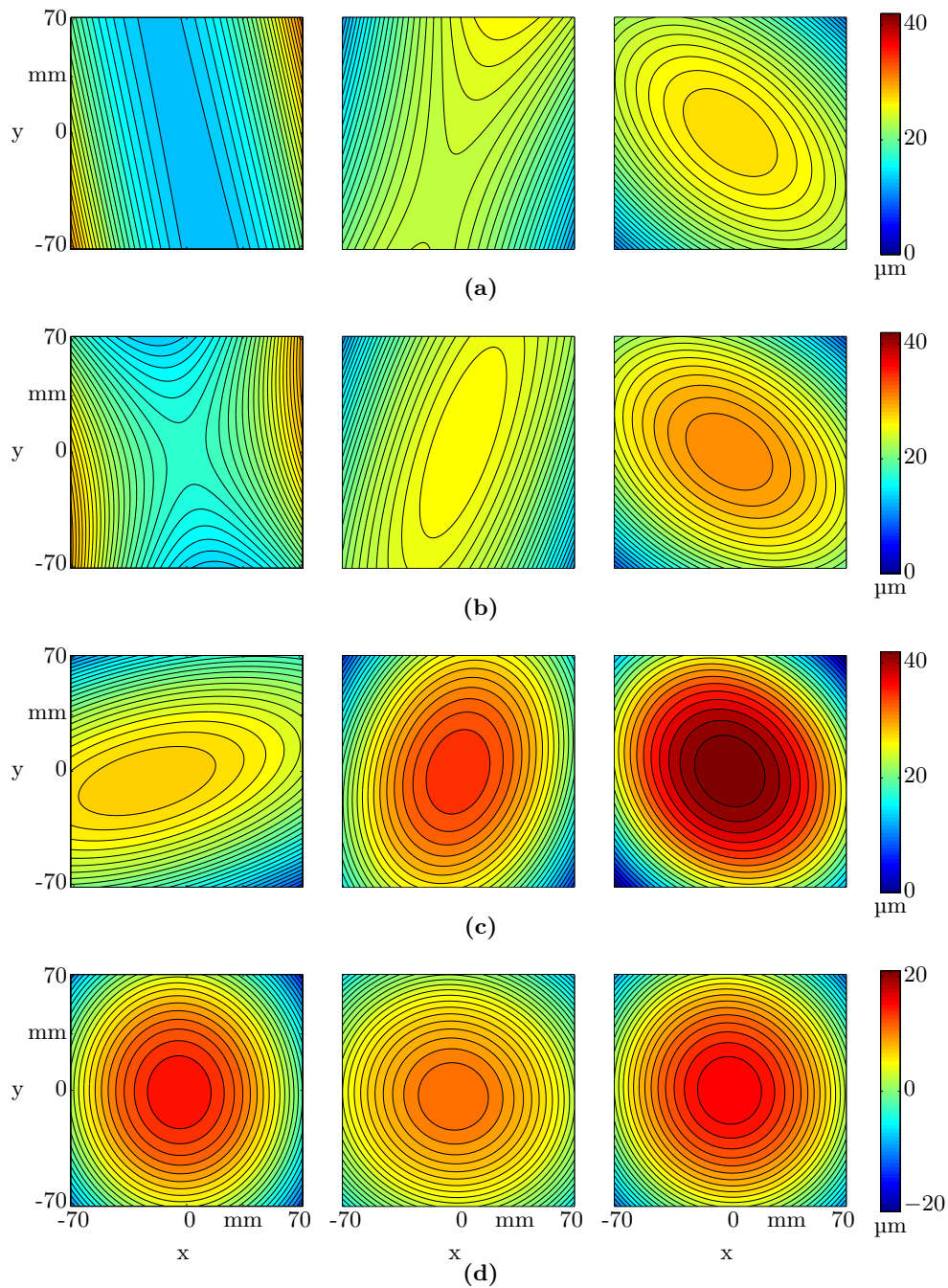


Abbildung 3.21: Konturgrafiken verschiedener Fittingpolynome $g^2(x, y)$ zweiten Grades für drei verschiedene Footprintproben in den drei Spalten. (a) Auf allen Punkten berechnete Fittingpolynome für die unbeschichteten Substrate mit unterschiedlich ausgerichteten zylindrischen, sattelförmigen und elliptischen Charakteristiken; (b) Nur auf den Randwerten der *unbeschichteten* Substrate berechnete Fittingpolynome – die Approximationen sind für den zentralen Bereich des Footprints ungenauer als diejenigen, die auf allen Punkten berechnet wurden; (c) Auf dem Referenzrand der *beschichteten*, thermisch verformten Proben approximierte Substratoberfläche, welche nun für alle Proben eine deutliche, nach oben gerichtete Wölbung aufweist; (d) Differenz zwischen den Fittings aus (c) und (a). Unabhängig von der initialen Charakteristik, Ausrichtung oder der Konvexität oder Konkavität verformen sich die Proben im Spritzprozess auf ähnliche Weise.

Methode 1

Die erste Methode approximiert den unbekanntem Verlauf der Substratgrenze als Extrapolation des unbeschichteten Referenzrandes R_i der beschichteten Probe, welcher in grüner Farbe in Abbildung 3.19 hervorgehoben ist. Dazu wird, wie im vorherigen Kapitel beschrieben, eine polynomielle Fläche zweiten Grades $f_i^R(x, y)$ auf die Randpunkte R_i gefittet (Abbildung 3.21c). Zur Vermeidung von Ungenauigkeiten werden dabei Ausreißer aus R_i , welche stellenweise durch nach der Reinigung des Probenrandes verbliebene Kleberanhaftungen entstanden sind, entfernt. Zu diesem Zweck wird zunächst ein vorläufiges Fitting berechnet und anschließend diejenigen Scanpunkte, die einen großen Fehler aufweisen, vor der Berechnung des finalen Fittings f_i^R aus R_i entfernt. Anhand des Fehlerhistogramms sind diese Ausreißer leicht zu identifizieren. Die Korrektur der Scandaten \mathcal{F}_i erfolgt schließlich durch die Subtraktion der auf diese Weise geschätzten Substratfläche von der originalen Punktwolke:

$$\tilde{\mathcal{F}}_i = \left\{ \left(x_j, y_j, z_j - f_i^R(x_j, y_j) \right)^T \mid \mathbf{x}_j = (x_j, y_j, z_j)^T \in \mathcal{F}_i \right\}. \quad (3.1)$$

Die Genauigkeit dieser Methode kann abgeschätzt werden, indem der Fehler der Randapproximation g_i^R des unbeschichteten Substrats betrachtet wird, da in diesem Fall der genaue Verlauf der Substratoberfläche auch in der Mitte bekannt ist. Nach Tabelle 3.2 beträgt dieser im Durchschnitt über alle Substrate $E_{AAD} = 2,30 \mu\text{m}$.

Methode 2

Die zweite Methode basiert auf der Feststellung, dass alle Footprints substrate unabhängig von ihrer initialen Verformungscharakteristik oder -richtung durch den thermischen Einfluss auf ähnliche Weise verformt werden. Deutlich wird dies in Abbildung 3.21d, welche für die drei exemplarisch betrachteten Footprints die Differenzen $f_i^R(x, y) - g_i^A(x, y)$ zwischen der jeweiligen Randapproximation des beschichteten und der auf allen Punkten berechneten Approximation des zugehörigen unbeschichteten Footprints zeigt. In allen acht Fällen ist in der Mitte eine Wölbung der Probe zur Beschichtungsseite zu beobachten. Diese ist für die betrachteten Proben – im Rahmen der Messgenauigkeiten – unabhängig von der Beschichtungsdauer von 5–10 s. Eine der Proben, welche während der Beschichtung rückseitig mit Druckluft gekühlt wurde, weist jedoch eine deutlich abweichende Durchbiegungscharakteristik auf und wird daher von dieser Methode ausgenommen. Wird nun für die übrigen Proben eine durchschnittliche Verformung

$$\bar{v}(x, y) = \frac{1}{7} \sum_{i=1}^7 \left(f_i^R(x, y) - g_i^A(x, y) \right) \quad (3.2)$$

definiert, so kann die Schicht-Substrat-Grenze eines Footprints i durch Addition $g_i^A(x, y) + \bar{v}(x, y)$ dieser Verformung auf das unbeschichtete Substrat geschätzt werden. Insgesamt ergibt sich damit das korrigierte Profil

$$\tilde{\tilde{\mathcal{F}}}_i = \left\{ \left(x_j, y_j, z_j - (g_i^A(x, y) + \bar{v}(x, y)) \right)^T \mid \mathbf{x}_j = (x_j, y_j, z_j)^T \in \mathcal{F}_i \right\}. \quad (3.3)$$

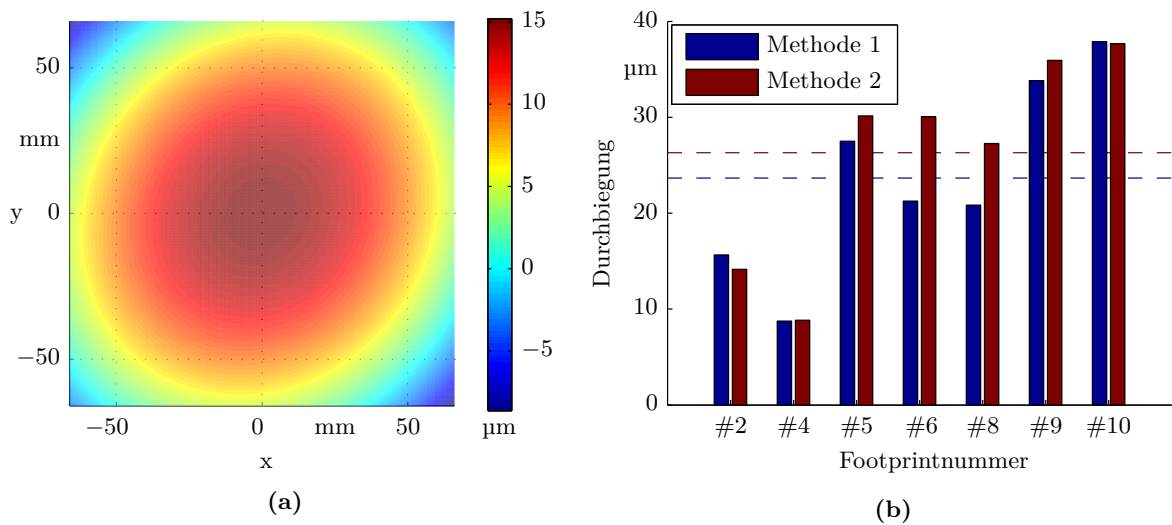


Abbildung 3.22: Thermische Verformungen: (a) Durchschnittliche vertikale Verformung der sieben Substrate dargestellt über der Substratfläche (Methode 2). (b) Mit den beiden unterschiedlichen Methoden festgestellte, aus dem Spritzprozess resultierende Substratdurchbiegungen bei sieben Footprints sowie deren Mittelwerte (gestrichelte Linien).

Die Genauigkeit dieser Methode hängt davon ab, wie gut $\bar{v}(x, y)$ die Verformung jedes Substrates wiedergibt. Ein Indiz hierfür ist die durchschnittliche absolute Abweichung jeder einzelnen Verformung $f_i^R(x, y) - g_i^A(x, y)$ in Gleichung (3.2) von $\bar{v}(x, y)$, welche für die vorliegenden Footprints im Mittel lediglich $1,1 \mu\text{m}$ (95 % der Werte kleiner als $2,8 \mu\text{m}$) beträgt. Zusätzlich muss der Fehler der Approximation der unbeschichteten Substratfläche g_i^A berücksichtigt werden, welche als Ausgangspunkt der Schätzung der Substratoberfläche dient und nach Tabelle 3.2 ca. $1,43 \mu\text{m}$ beträgt. Demgegenüber hängt die Genauigkeit der ersten Methode im Wesentlichen von der Genauigkeit der randbasierten Fittings f_i^R ab, welche für die unbeschichteten Substrate $2,3 \mu\text{m}$ beträgt. Insgesamt ist die erste Methode daher anfälliger für Messfehler auf dem Rand der beschichteten Proben, beispielsweise aufgrund von Kleberanhaftungen, und die zweite Methode für starke Abweichungen von der Durchschnittsverformung \bar{v} , welche unter anderem aus einem abweichenden Temperaturverlauf im Prozess resultieren können. Aus rein praktischer Sicht ist die erste Methode deutlich einfacher zu realisieren, da hierfür die unbeschichteten Proben nicht gescannt werden müssen.

Abbildung 3.22a zeigt die nach der zweiten Methode durchschnittlich angewandte Korrektur, welche qualitativ von der nach der ersten Methode berechneten Korrektur kaum zu unterscheiden ist. Die von beiden Methoden berechneten und korrigierten Durchbiegungen, also die Intervallbreiten der addierten Korrekturterme, für die sieben mit beiden Methoden korrigierten Footprintproben sind in Abbildung 3.22b dargestellt. Es ist zu erkennen, dass die berechneten Korrekturen beider Methoden ähnlich ausfallen, wobei die zweite Methode mit im Durchschnitt $26,3 \mu\text{m}$ gegenüber $23,7 \mu\text{m}$ bei der ersten Methode etwas stärkere Durchbiegungen schätzt. Zur Kontrolle der nach der Verformungskompensation resultierenden Schichtdickenverläufe wurden diese für einen der Footprints (#8) und eine der Linearbahnproben

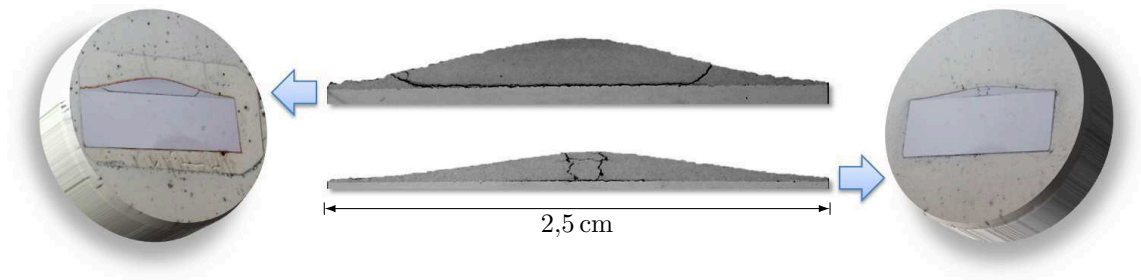


Abbildung 3.23: Darstellung der aufgetrennten, eingebetteten, geschliffenen und polierten Proben des Footprints #8 (links) und des Linearbahnprofils (rechts) sowie ihrer Mikroskopaufnahmen (mitte).

(vergleiche Kapitel 3.8) durch Anfertigung eines Querschliffs überprüft, wobei die im folgenden Kapitel 3.7 zur Vermessung der Schichtdicke entwickelten Algorithmen eingesetzt wurden (Abbildung 3.23). Für den ca. 2,5 cm langen Querschliff des Footprints ergibt sich nach Filterung des Scan- und Oberflächenrauschens mit einem 1 mm breiten gleitendem Mittelwertfilter eine durchschnittliche absolute Abweichung von $35,4\ \mu\text{m}$ vor und $29,7\ \mu\text{m}$ nach der Kompensation der thermischen Verformung – 1,9 % respektive 1,59 % der maximalen Schichtdicke. Es konnte also tatsächlich eine Verbesserung festgestellt werden, welche jedoch in diesem Fall – entsprechend der ohnehin geringen Durchbiegung der 30 mm starken Substrate – nicht sehr groß ausfällt. Der Unterschied zwischen den beiden Kompensationsmethoden ist hier im Hinblick auf die Abweichung von der Querschliffmessung marginal, so dass die Entscheidung zur Verwendung der ersten Methode in diesem Fall vorwiegend auf Basis der einfacheren Einsetzbarkeit beruht. Im allgemeinen Fall ist eine Entscheidung für eines der beiden Verfahren jedoch neu zu treffen, wobei insbesondere die Gleichförmigkeit der Verformungen verschiedener Proben kritisch geprüft werden sollte. Kann diese für den vorliegenden Prozess nicht gewährleistet werden, ist Methode 1 vorzuziehen.

3.6.4 Konvertierung für die Nutzung in der Simulation

Zur Nutzung eines ausgerichteten und korrigierten Footprints in der Simulation muss dieser als regulär abgetastetes Höhenfeld in eine Textur geladen werden (Kapitel 5.3.3). Zu diesem Zweck wird zunächst eine Delaunay-Triangulation [9] aller Punkte der verformungskorrigierten Punktwolke in der x - y -Ebene berechnet. Auf Basis dieser Triangulation können Höhenwerte des Profils an beliebigen Stellen (x, y) linear aus den Werten der drei Eckpunkte des Dreiecks, in dem (x, y) liegt, interpoliert werden. Eine Abtastung an den Stellen eines regulären Gitters $(i \cdot \Delta x, j \cdot \Delta y)$ in der x, y -Ebene liefert nun das für die Simulation benötigte Format. Die neu abgetasteten Footprints \mathcal{F}_i können somit zum einen nach wie vor als – nun über einem regulären Gitter angeordnete – Punktmengen $\{\mathbf{x}_j\}$ verstanden werden, zum anderen aber auch als Funktionen $\mathcal{F}_i(x, y)$, deren Funktionswerte an der Stelle (x, y) wiederum durch (bilineare) Interpolation benachbarter Punkte aus \mathcal{F}_i ermittelt werden.

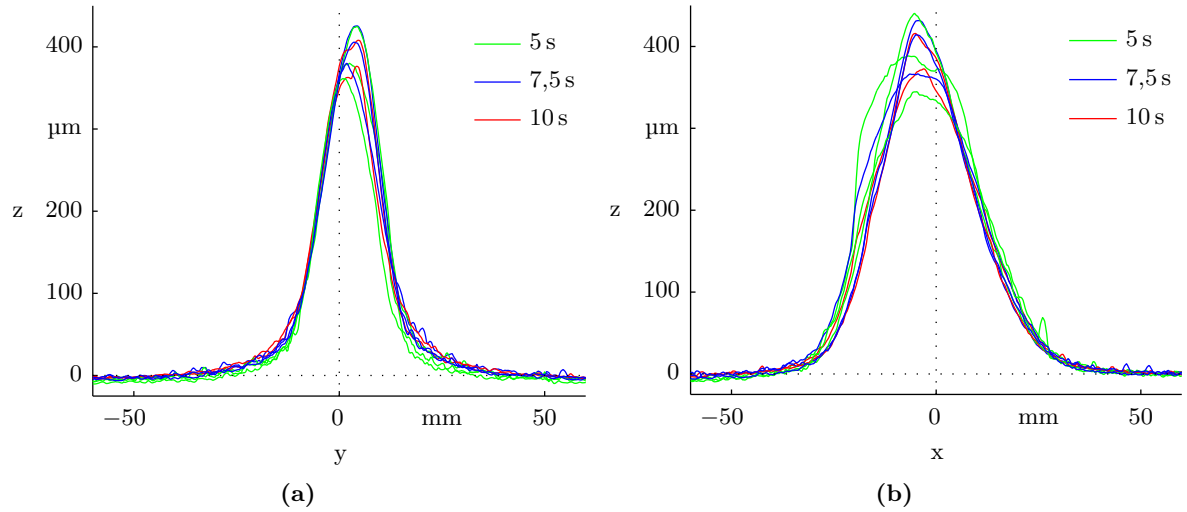


Abbildung 3.24: Querschnitte dreidimensionaler Footprintprofile entlang der im Versuch senkrechten y -Achse (a) bzw. der waagerechten x -Achse (b). Enthalten sind je drei Footprints mit 5 s, drei mit 7,5 s und zwei mit 10 s Spritzdauer. Die Schichtdicke dieser auf eine Sekunde Spritzdauer normierten Einzelproben weist eine deutliche Streuung auf. Ferner zeigen insbesondere die waagerechten Querschnitte verschieden stark ausgeprägte Wölbungen der Profile. Diesen Effekten liegen jedoch Prozessfluktuationen zugrunde, welche keine systematische Abhängigkeit von der Spritzdauer erkennen lassen. Es kann daher von einem näherungsweise homothetischen Schichtwachstum ausgegangen werden.

Um die Genauigkeit eines in der Simulation verwendeten Footprints zu steigern, ist es sinnvoll, diesen Footprint durch Mittelwertbildung aus Profilen mehrerer Einzelmessungen zu erzeugen. Dadurch werden sowohl das in den Einzelmessungen enthaltene Scanrauschen als auch die in den Footprints enthaltenen Prozessfluktuationen globaler (schwankende Auftragsrate) und lokaler (anhaftende Partikelagglomerationen) Art gedämpft. Unter Berücksichtigung der verschiedenen Spritzzeiten kann aus einer Menge Θ von Footprints ein mittlerer Footprint

$$\bar{\mathcal{F}}(x, y) = \frac{1}{|\Theta|} \sum_{i \in \Theta} \frac{1}{t_i} \mathcal{F}_i(x, y) \quad (3.4)$$

erzeugt werden, welcher auf eine Sekunde Spritzdauer normiert ist. Dies setzt zunächst voraus, dass der bei der Lasereinmessung markierte Arbeitspunkt für alle Proben identisch ist, was in diesem Fall durch die verwendete Probenaufnahme sichergestellt wird. Weiterhin stellt sich die Frage, welche der Proben in einen mittleren Footprint einfließen sollten beziehungsweise ob und wie die Qualität der Einzelmessungen beurteilbar ist.

Als ein die Qualität beeinflussendes Kriterium könnte die Spritzdauer dienen, da bei Footprints mit größerer Spritzdauer Prozessfluktuationen durch die längere zeitliche Mittelung besser ausgeglichen werden und aufgrund der höheren Schichtdicke ein geringerer relativer Messfehler in der Profilmessung erreicht wird. Andererseits besteht mit zunehmender Spritzdauer auch die Gefahr einer Veränderung der Profilform aufgrund einer stärkeren Erwärmung der Probe oder eines möglichen Einflusses des höheren und damit auch steileren Footprintpro-

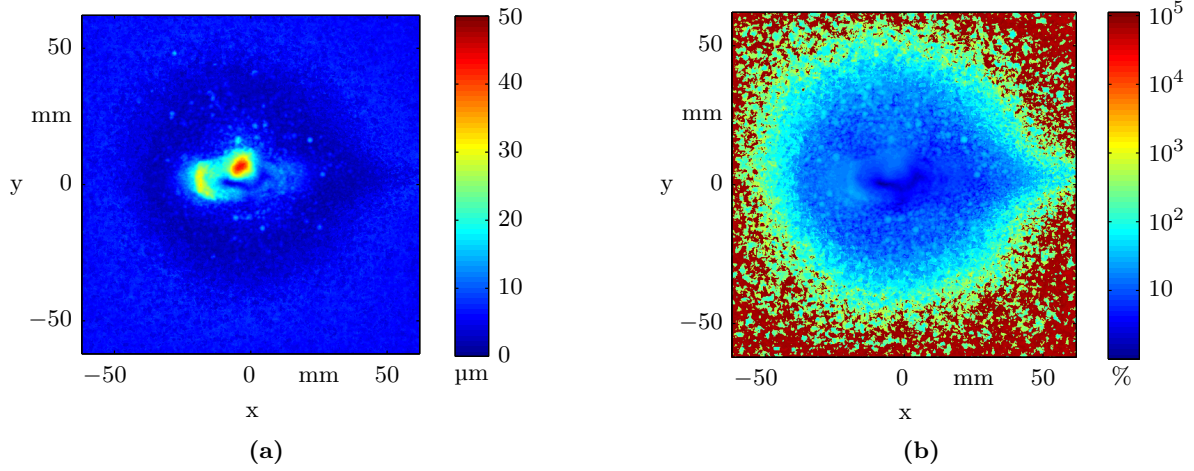


Abbildung 3.25: : Durchschnittliche (a) absolute und (b) relative Abweichung der einzelnen Profile $\mathcal{F}_i \in \Theta$ vom Durchschnittsfootprint $\bar{\mathcal{F}}$. Der starke Anstieg des relativen Fehlers (logarithmische Darstellung) zum Rand ist in der stark abfallenden Schichtdicke begründet.

fils auf die Partikelhaftung bzw. das Ablagerungsverhalten. Eindeutige Indikatoren für eine zur Spritzdauer proportionale oder antiproportionale Qualität scheint es in diesem Fall jedoch nicht zu geben, wie aus den Querschnitten durch die acht Einzelprofile entlang der y- bzw. x-Achse in den Abbildungen 3.24a respektive 3.24b hervorgeht. Zwar scheinen die Footprints mit 5 s Spritzdauer an der oberen und unteren Flanke (ca. ± 20 mm in Abbildung 3.24a) etwas dünner zu sein als die übrigen Footprints, die Maximalhöhe oder die Halbwertsbreite (Breite bei halber Maximalhöhe) ist jedoch unabhängig von der Spritzdauer. So haben beispielsweise sowohl die niedrigste als auch die höchste normierte Footprintprobe eine Spritzdauer von 5 s. Da sich auch die Volumina der normierten Footprintproben nicht proportional oder antiproportional zur Spritzdauer verhalten, kann des Weiteren von einem Fluktuationen unterworfenen, aber insgesamt näherungsweise homothetischen Schichtwachstum im betrachteten Spritzdauerbereich ausgegangen werden. Dies bedeutet insbesondere, dass an dieser Stelle Footprintproben verschiedener Spritzdauer in die Bildung eines mittleren Footprints einbezogen werden können. Daher werden an dieser Stelle alle acht Proben, deren Verformungen mit der ersten in Gleichung (3.1) beschriebenen Methode kompensiert wurden, zur Bildung eines mittleren Footprints eingesetzt. Abbildung 3.25a zeigt die Visualisierung der gemittelten absoluten Abweichung

$$e_{\text{AAD}}(x, y) = \frac{1}{|\Theta|} \sum_{i \in \Theta} \left| \frac{1}{t_i} \mathcal{F}_i(x, y) - \bar{\mathcal{F}}(x, y) \right|$$

der einzelnen Proben von $\bar{\mathcal{F}}$, welche im Maximum $43,3 \mu\text{m}$ beträgt. Anhand der relativen Abweichungen

$$e_{\text{rel}}(x, y) = \left(e_{\text{AAD}}(x, y) / \bar{\mathcal{F}}(x, y) \right) \cdot 100\% \quad (3.5)$$

der an der Bildung des durchschnittlichen Footprints beteiligten Einzelprofile in Abbildung 3.25b ist zu erkennen, dass der relative Fehler aufgrund der nach außen fast auf Null

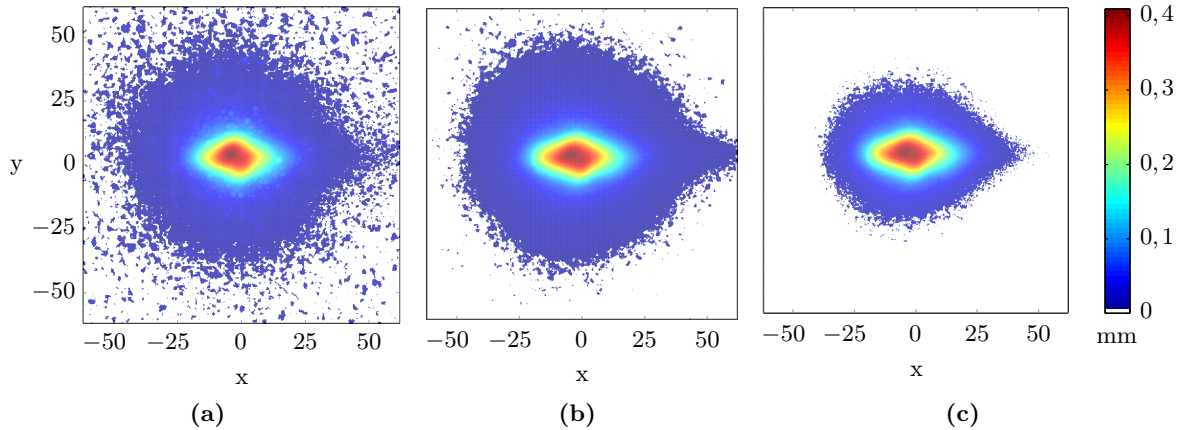


Abbildung 3.26: Ergebnisse der unterschiedlichen Methoden zur Bestimmung der unbeschichteten Bereiche des Footprints. Das einfache Schwellwertverfahren mit Grenzwert γ auf einer Einzelprobe (a) liefert keine guten Ergebnisse, da γ im sandgestrahlten Bereich bis an den Rand der Probe häufig überschritten wird. (b) Bei Mittelwertbildung mehrerer Proben kann durch die Anwendung des Grenzwertes $\bar{\gamma}$ nach Gleichung (3.6) auf das mittlere Footprintprofil ein besseres Ergebnis $\bar{\mathcal{F}}_{\text{loose}}$ erzielt werden. (c) Ein noch enger gefasstes Profil ergibt sich durch die Beschränkung auf den Bereich, in dem alle Proben den Grenzwert γ überschreiten ($\bar{\mathcal{F}}_{\text{tight}}$). In beiden Fällen, (b) und (c), ist die von der Bewegung der Spritzpistole über das Substrat verursachte leichte Ausbuchtung an der rechten Seite zu erkennen.

abnehmenden Schichtdicke extrem ansteigt und ab einem gewissen Punkt nur noch von den hochfrequenten Anteilen der gesandstrahlten Oberfläche sowie vereinzelt anhaftenden Partikeln bestimmt wird. Daher sind für eine Verwendung des Footprints in der Simulation unbeschichtete Bereiche zu identifizieren und auf Null zu setzen.

Wie in Kapitel 3.6.2 bereits dargelegt wurde, wird die Profilhöhe sandgestrahlter Bereiche durch den taktilen Scanvorgang um durchschnittlich $\gamma = 76,6 \mu\text{m}$ überschätzt, weshalb Bereiche mit geringeren Profilhöhen als unbeschichtet angenommen werden können. Da es sich bei diesem Grenzwert nur um einen Mittelwert handelt, welcher auch an vielen unbeschichteten Stellen überschritten wird, liefert die Anwendung auf eine einzelne Probe kein gutes Ergebnis (Abbildung 3.26a). Eine bessere, in weniger Teile zerfallende Segmentierung der beschichteten Bereiche lässt sich für die aus mehreren Einzelprofilen Θ kombinierten Footprints erzielen, indem die Segmentierung auf dem gemittelten Footprint berechnet wird. Dabei werden diejenigen Bereiche als unbeschichtet angenommen, die im Mittel der einfließenden Footprints eine kleinere Profilhöhe als γ aufweisen. Da jeder der in die Mittelung einfließenden Footprints auf eine Sekunde Spritzdauer normiert wird, muss auch der Grenzwert γ entsprechend Gleichung (3.4) normiert werden. Es können daher in dem betrachteten Fall alle Punkte $\mathbf{x}_j \in \bar{\mathcal{F}}$ als unbeschichtet angenommen und für diese $z_j = 0$ gesetzt werden, deren Profilhöhe z_j kleiner als

$$\bar{\gamma} = \sum_{i \in \Theta} \frac{1}{t_i} \cdot 76,6 \mu\text{m} \quad (3.6)$$

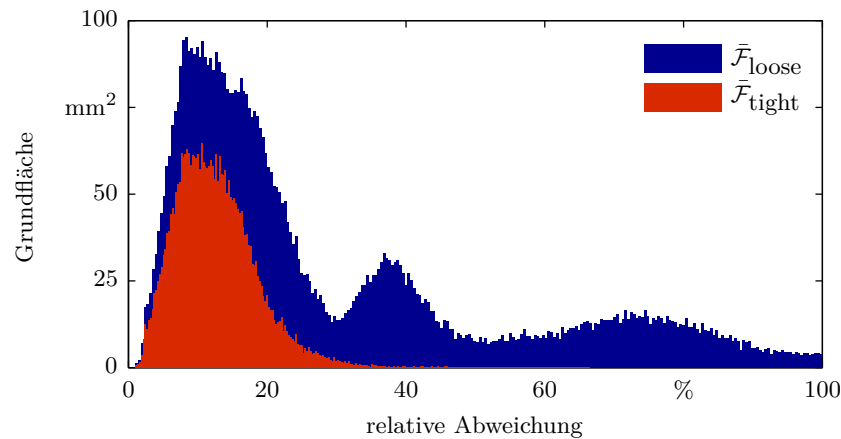


Abbildung 3.27: Histogramm der relativen Abweichungen der Einzelversuche von den durchschnittlichen Footprints \bar{F}_{loose} und \bar{F}_{tight} als Indikator für die Reproduzierbarkeit beziehungsweise die Prozessstabilität beim Lichtbogenspritzen. Die Abweichungen wurden auf Basis der als beschichtet klassifizierten Bereiche (Abbildung 3.26) berechnet.

ist. Alternativ kann auch nur der Teil der Oberfläche als unbeschichtet angenommen werden, in dem *keines* der Einzelprofile den Grenzwert γ überschreitet. Dadurch wird der beschichtete Bereich enger gefasst. Die beiden resultierenden Footprints \bar{F}_{loose} und \bar{F}_{tight} sind in Abbildung 3.26 dargestellt. Die durchschnittlichen relativen Abweichungen aller Einzelfootprints von \bar{F}_{loose} und \bar{F}_{tight} sind in Abbildung 3.27 als Histogramm dargestellt und liefern ein wichtiges Indiz für die Streuung des Beschichtungsprozesses und somit auch für die zu erwartende Genauigkeit einer Simulation auf Basis dieser Daten. Im Mittel beträgt die relative Abweichung 29,5 % für \bar{F}_{loose} und 12,6 % für \bar{F}_{tight} . Die beiden Häufungen der Abweichungen bei ca. 40 % und 75 % treten in den äußeren Bereichen von \bar{F}_{loose} auf, in denen die durchschnittliche Schichtdicke bereits in die Größenordnung der Einzelpartikeldicken abgesunken ist.

Mit der erfolgten Ausrichtung der Profile, der Kompensation thermischer Verformungen und der Mittelwertbildung zur Unterdrückung von Prozessfluktuationen sind die wichtigsten Schritte der Nachbearbeitung dreidimensionaler Profile durchgeführt und die Profile können im Rahmen der Simulation verwendet werden. Als zusätzliche Eingaben für die Simulation werden dafür noch die Spritzdauer, der Spritzabstand der Footprints sowie der durch die Lasereinmessung der Proben bestimmte Arbeitspunkt für die Simulation benötigt. Letzterer ist für alle Proben identisch und ergibt sich als Schnittpunkt der Laserkreuzachsen, welche wiederum durch die an den Probenrändern aufgebrauchten Markierungen bestimmt sind (Kapitel 3.4.1). Eine weitere Steigerung der Genauigkeit der Footprintprofile und eine effiziente Rauschunterdrückung kann durch die zusätzliche Anwendung der modellbasierten Footprintoptimierung in Kapitel 3.8 oder durch die in Kapitel 3.9 beschriebene Modellierung des Profils als Summe von Gaußfunktionen erzielt werden.

3.7 Nachbearbeitung zweidimensionaler Profile

Im Gegensatz zu den dreidimensionalen Profilen, welche bereits nach dem Scanprozess als potentielle Schichtdicken vorliegen und im Rahmen der Nachbearbeitung lediglich korrigiert werden, müssen die zweidimensionalen Profile noch aus den mit dem digitalen Weißlichtmikroskop erfassten Bildern extrahiert werden. Dabei entsteht die Gesamtaufnahme einer aufgetrennten, eingebetteten, geschliffenen und polierten Schichtprobe mit einem Durchmesser von maximal 30–40 mm aus mehreren (bis zu 30–40) Einzelaufnahmen, welche von der Mikroskopiesoftware zu einem Bild zusammengesetzt werden. Eine Gesamtaufnahme kann dabei mehrere 100 Megapixel groß sein. Die Messung der Profile in diesen Bildern konnte bisher nur durch manuelles Einfügen und Tabellarisieren einer großen Anzahl einzelner Messbalken mit Hilfe der Mikroskopiesoftware durchgeführt werden. Um den damit verbundenen Arbeitsaufwand von ca. 30–45 Minuten pro Probe zu reduzieren und eine objektivere und hochauflösendere Messung zu ermöglichen, wurde ein Algorithmus entwickelt und als eigenständiges Softwaretool implementiert, welches diesen Messvorgang weitestgehend automatisiert. Der Algorithmus arbeitet in mehreren Schritten: Zunächst wird das Bild durch Anwendung lokaler Operatoren segmentiert, um den Bildbereich, der die Beschichtung darstellt, zu ermitteln. Diese Segmentierung wird anschließend mittels morphologischer Filterung korrigiert und skelettiert (ausgedünnt) [64], um eine zentral entlang der Beschichtung verlaufende Mittelachse zu ermitteln. Diese Mittelachse dient im letzten Schritt als Grundlage zur Berechnung des Verlaufs der Schichtdicke senkrecht zu dieser Achse. Die einzelnen Schritte des Algorithmus werden im Folgenden detailliert beschrieben.

1. Segmentierung

Um zunächst den Bildbereich zu identifizieren, welcher die Beschichtung darstellt, wird die Tatsache ausgenutzt, dass dieser Bereich aufgrund der vielen Grenzen zwischen den einzelnen Splats der Beschichtung wesentlich hochfrequenter Bildanteile enthält als das Substrat (helle Bereiche in Abbildung 3.28a und 3.28b) oder die Einbettmasse (dunkle Bereiche). Diese Splatgrenzen werden zunächst mit Hilfe des Canny-Kantendetektors [21] ermittelt, wobei insbesondere auch die Grenzen zwischen Beschichtung und Einbettmasse bzw. Beschichtung und Substrat als Kanten erkannt werden. Das Ergebnis ist ein Binärbild, in welchem die Kantenpixel weiß und alle anderen schwarz sind (Abbildung 3.28c). Da die Kantenverläufe, welche die Beschichtung beranden, in der Regel noch viele Lücken enthalten und Algorithmen zum Kantenschließen bei derart vielen Kantenpixeln nicht ohne Weiteres eingesetzt werden können, wird zur weiteren Segmentierung ein einfacher flächenbasierter Ansatz gewählt. Dazu wird das Kantenbild mit einem Gaußfilter tiefpassgefiltert, um die vielen weißen dicht beieinander liegenden Kantenpixel ineinander verlaufen zu lassen (Abbildung 3.28d). Dadurch entsteht an der Stelle der Beschichtung ein inhomogener heller Bereich, welcher sich mit einem anschließenden Schwellenwertfilter leicht extrahieren lässt. Durch den Tiefpaßfilter werden dabei dünne linienartige Kantenstrukturen, wie zum Beispiel die Kanten der

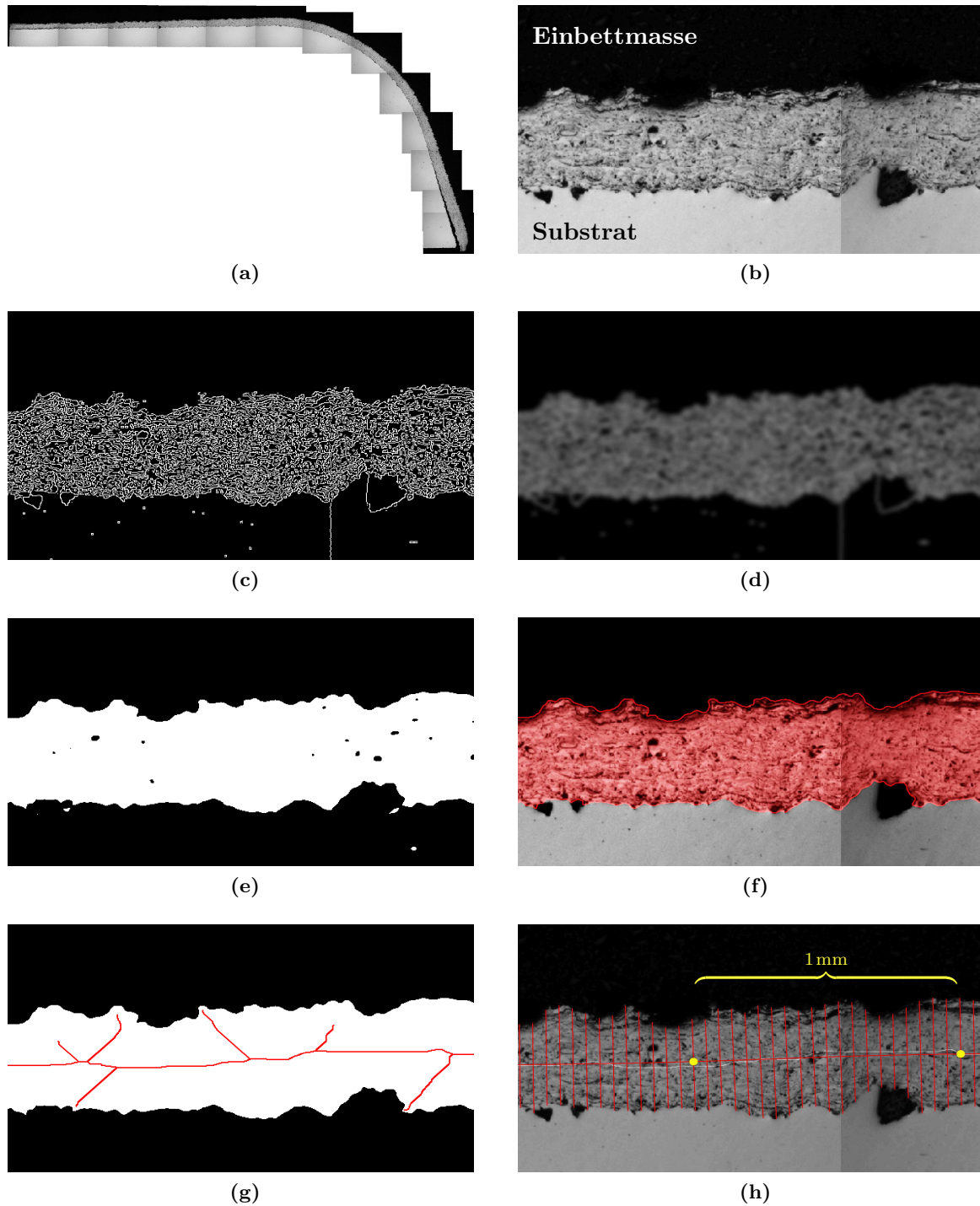


Abbildung 3.28: Darstellung der Zwischenergebnisse des entwickelten Algorithmus zur Schichtdickenmessung. (a) Gesamtaufnahme der Probe - zusammengesetzt aus mehreren Einzelbildern. (b) Unbearbeiteter Ausschnitt. (c) Ausschnitt nach der Kantendetektion mittels Canny-Algorithmus [21]. (d) Ausschnitt nach der gaußschen Tiefpassfilterung. (e) Ausschnitt nach Anwendung eines Schwellenwerts. (f) Detektiertes Schichtsegment nach der morphologischen Bereinigung. (g) Ermitteltes Skelett des Schichtsegments mit unerwünschten Verästelungen. (h) Aus dem reduzierten Skelett gewonnene Mittelachse mit dazu senkrechten Messbalken im Abstand 50 µm; gelbe Punkte markieren 1 mm-Abstände.

großen Poren oder die aufgrund von Belichtungsunterschieden detektierten Kanten an den Grenzen zusammengefügt Einzelbilder, wie sie in Abbildung 3.28c zu sehen sind, erfolgreich entfernt, da ihre Helligkeitsdichte zu gering ist, um nach der Tiefpassfilterung über dem Schwellenwert zu liegen.

In Abbildung 3.28e ist zu erkennen, dass dennoch mehrere - im gesamten Querschliffbild bis zu mehreren hundert - weiße Bereiche an den Stellen übrig bleiben, in denen im Originalbild unregelmäßige Strukturen im Substrat oder der Einbettmasse auftreten. Da in der Regel der weiße Bereich mit der größten Fläche der gesuchten Beschichtung entspricht, kann dieser ermittelt werden, indem alle Pixel gemäß ihrer Zugehörigkeit zu Zusammenhangskomponenten nummeriert und anschließend gezählt werden. Dabei kommt ein zweiphasiger Linearzeitalgorithmus zur Etikettierung von Zusammenhangskomponenten (engl.: connected component labeling) zum Einsatz, welcher zum Beispiel in [164] und [135] beschrieben ist. Nach der Etikettierung wird nur die größte Zusammenhangskomponente behalten, alle übrigen repräsentieren unerwünschte weiße Fehlstellen außerhalb der eigentlichen Beschichtung und werden entfernt (geschwärzt). Für die folgende Berechnung der Mittelachse der Beschichtung müssen des Weiteren auch die aufgrund von inneren Poren innerhalb der Schicht verbliebenen Löcher (Abbildung 3.28e) entfernt werden. Hierfür kann die Tatsache ausgenutzt werden, dass der Bereich außerhalb der Schicht bereits von weißen Fehlstellen bereinigt wurde. Mittels eines einfachen Floodfill-Algorithmus [64] wird der äußere, schwarze Bereich markiert, wobei eine zuvor durchgeführte Erweiterung des Bildbereichs um einen ein Pixel breiten (schwarzen) Rand sicherstellt, dass alle eventuell disjunkten Randbereiche erfasst werden. Alle unmarkierten Pixel gehören dann zu der bereits segmentierten Beschichtung oder zu entfernenden inneren Löchern. Abbildung 3.28f zeigt den segmentierten Schichtbereich, welcher anschließend zu vermessen ist.

2. Bestimmung der Mittelachse

Da das Schichtprofil idealerweise stets senkrecht zur Oberfläche des Bauteils bzw. zur Verlaufsrichtung der Schicht gemessen wird, wird im nächsten Schritt die Mittelachse als Verlaufsrichtung der Schicht berechnet. Dazu wird das zuvor ermittelte Schichtsegment in einem ersten Schritt, unter Verwendung des morphologischen Algorithmus von Zhang und Suen [182], skelettiert. Dieser Algorithmus entfernt in einem iterativen Verfahren solange nach und nach Konturpunkte des ausdünnenden Bereichs, deren 3×3 -Pixelnachbarschaft eine Reihe von Kriterien erfüllt, bis nur noch ein Skelett aus Linien mit einer Breite von einem Pixel übrig bleibt. Neben der Bedingung, dass nur Konturpixel entfernt werden, wird dabei gewährleistet, dass das Gebiet nicht zerfällt und keine Endpixel der Skelettlinien entfernt werden. Aufgrund der potentiell sehr großen Eingabebilder und der Prüfung mehrerer Bedingungen auf der Umgebung jedes Pixels in jeder Iteration stellt dieser Algorithmus trotz vollständig paralleler Implementierung unter Verwendung von OpenMP (Open Multi-Processing) [24] den zeitlichen Flaschenhals der gesamten automatischen Schichtberechnung dar. Mit einem Zeitbedarf

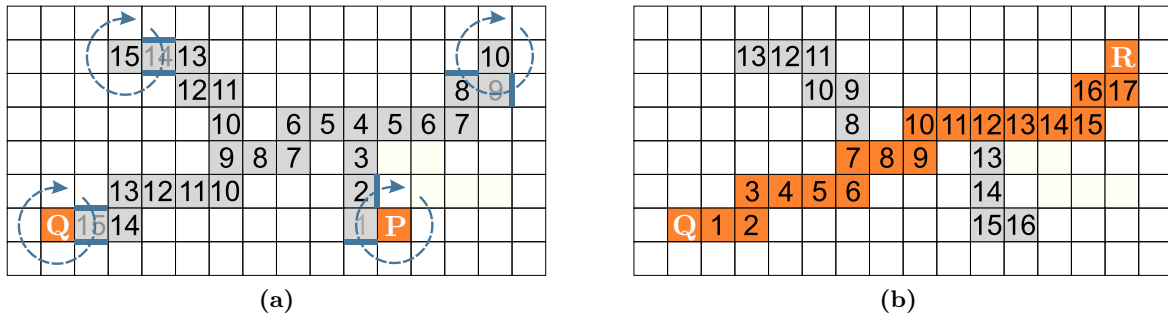


Abbildung 3.29: Berechnung des längsten Weges im Skelettgraph. (a) Endpunkte des Skeletts sind genau diejenigen Pixel, deren im Kreis enumerierten Nachbarn genau zwei Schwarz-Weiß-Wechsel aufweisen (blaue Balken). Ausgehend von einem beliebigen Endpunkt P wird zunächst mittels Flood-Fill-Algorithmus (Zahlen) der am weitesten entfernte Endpunkt Q bestimmt. (b) Ausgehend von Q wird dieses Verfahren wiederholt, um den längsten Pfad des Skeletts von Q nach R zu bestimmen.

von ca. 25 s bei dem in Abbildung 3.28a präsentierten Beispiel der Größe 10000×5200 auf einem Desktop PC mit Core-i7 2600 (3,4 GHz) und 8-facher Parallelisierung ist dieser jedoch akzeptabel und insbesondere im Vergleich zur manuellen Segmentierungsdauer vernachlässigbar. Abbildung 3.28g zeigt einen Teil des resultierenden Skeletts, welches aufgrund der äußeren Kontur des Schichtsegments eine Vielzahl an – in diesem Fall unerwünschten – Verästelungen enthält. Mit Hilfe dieses Skeletts kann nun jedoch eine gute Näherung für die Mittelachse der Schicht als Verbindung derjenigen zwei Endpunkte des Skeletts gefunden werden, welche den größten Abstand voneinander haben. Es wird also der längste Pfad zwischen zwei der Endpunkte des entstandenen kreisfreien Graphen gesucht.

Eine wesentliche Schwierigkeit ergibt sich hier zunächst dadurch, dass dieser Graph bzw. Baum lediglich als Pixelbild vorliegt und seine logische Struktur nicht bekannt ist – es existieren keine Knoten oder Kantenlisten. Um Graphenalgorithmien auf der logischen Baumstruktur des Skeletts anwenden zu können, müssten also zunächst alle End- und Kreuzungspunkte ermittelt werden. Dies ist jedoch insbesondere für die Kreuzungspunkte nicht trivial, da hier viele verschiedene Konfigurationen auf Pixelebene auftreten können. Stattdessen werden in einem ersten Schritt zur Bestimmung des längsten Weges lediglich die Endpunkte des Skeletts als diejenigen Pixel identifiziert, deren acht im Kreis enumerierten direkten Nachbarpixel genau zwei schwarz-weiß Wechsel aufweisen (Abbildung 3.29a). Diese Charakterisierung ist eindeutig, da Pixel in der Mitte eines Pfades vier und Kreuzungspunkte mehr als vier Wechsel aufweisen. Von einem beliebigen dieser Endpixel ausgehend wird anschließend mit einem Flood-Fill-Ansatz die Distanz aller Skelettpixel zu diesem Startpixel P ermittelt. Anschließend wird der von P am weitesten entfernte Skelettendpunkt Q gewählt und das Verfahren wiederholt. Der Weg von Q zu dem am weitesten von Q entfernten Endpunkt R ist dann der längste Weg im Skelett und somit eine erste Näherung für die gesuchte Mittelachse der Beschichtung

(Abbildung 3.29b). Dieses Vorgehen entspricht im Wesentlichen dem Algorithmus von Dijkstra zum Finden des *längsten* Weges in einem Baum [161], wobei jeder Pixel des Skeletts als Knoten interpretiert und die makroskopische Konnektivitätsstruktur des Skeletts nie explizit berechnet wird. Ein Beweis für die Korrektheit findet sich in [17]. Um für die Messung der Schichtdicken senkrecht zum Verlauf der gefundenen Mittelachse nicht zu stark variierende Richtungen zu erhalten, wird die Mittelachse nicht direkt durch den gefundenen längste Pfad repräsentiert, sondern durch eine approximierende Splinekurve ersetzt [118].

3. Profil- und Porositätsmessung

Die in Schritt zwei ermittelte Splinekurve wird nun in frei definierbaren Abständen regelmäßig unterteilt, um anschließend an diesen Unterteilungspunkten die Dicke des zuvor segmentierten Schichtbereichs senkrecht zur Splinekurve zu vermessen. Diese Messungen sind in Abbildung 3.28h durch rote Messbalken veranschaulicht. Für das im Rahmen dieser Arbeit verwendete Mikroskop wird zur Umrechnung der Längen von Pixeln in μm der von der Mikroskopiesoftware im Bild eingebettete Maßstabsbalken automatisch erkannt. Die ermittelten Schichtdicken, die zugehörigen Messbalkenpositionen entlang der Oberfläche sowie die Bildkoordinaten werden von der im Rahmen dieser Arbeit entwickelten Messsoftware tabellarisch exportiert. Des Weiteren können entlang der Messbalken Richtwerte für die Porosität des Schichtgefüges gemessen und exportiert werden, falls ein Helligkeitsgrenzwert zur Abgrenzung von (dunkleren) Poren von der übrigen Schicht angegeben wird. Die Porosität wird dabei entlang jedes Messbalkens als prozentualer Anteil aller Pixel des originalen Schliiffbildes berechnet, deren Helligkeit unterhalb des gegebenen Schwellenwertes liegt.

Abbildung 3.30 zeigt beispielhaft für die in Abbildung 3.28a dargestellte Probe die Ergebnisse einer manuellen und einer automatisierten Messung im Vergleich. Abgesehen von der deutlich geringeren Messzeit – ca. zwei Minuten, im Vergleich zu 30–45 Minuten für die manuelle Messung – liefert die automatische Messung auch wesentlich höher aufgelöste Daten. Diese automatisch generierten Messdaten geben den tatsächlichen Profilverlauf genauer wieder als die Messwerte einer manuellen Vermessung, deren Messpunkte, aufgrund des menschlichen Faktors bei der Auswahl dieser Punkte, hin und wieder an Extremstellen des Schichtprofils gesetzt werden und somit den lokalen Verlauf der Schichtdicke nicht korrekt wiedergeben.

3.8 Nachbearbeitung durch modellbasierte Optimierung

Die Footprintprofile, welche mittels der in den Kapiteln 3.4 bis 3.6 beschriebenen Verfahren ermittelt werden können, werden als Kern der in Kapitel 5 beschriebenen Auftragssimulation eingesetzt. Dabei ergeben sich Abweichungen zwischen den durch die Simulation berechneten Schichtprofilen und den in identischen Versuchen experimentell ermittelten Profilen. Mögliche Ursachen für diese Abweichungen sind:

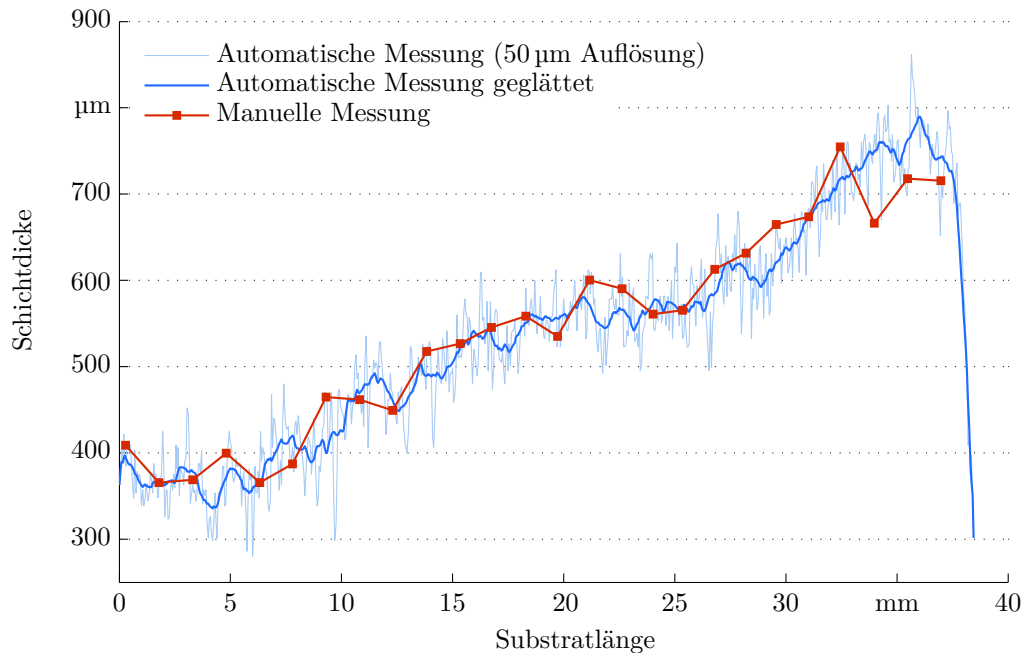


Abbildung 3.30: Manuell gemessenes und automatisch ermitteltes Schichtprofil der Probe aus Abbildung 3.28a im Vergleich. Die manuellen Messpunkte weisen ungleichmäßige Abstände zueinander auf und sind gelegentlich an nicht repräsentativen Extrempunkten platziert. Dies gilt beispielsweise für die Messpunkte an den Stellen $x \approx 9$ mm, $x \approx 22$ mm sowie $x \approx 34$ mm. Die automatisch berechneten Werte wurden in Abständen von $50 \mu\text{m}$ ermittelt und anschließend geglättet (Gaußfilter mit 1 mm breitem Kern).

- **Prozessfluktuationen**

Fluktuationen von Strom, Spannung, Gasdruck oder des Drahtvorschubs beim Lichtbogenspritzen sowie der Partikelförderrate bei anderen Verfahren können einen unregelmäßigen Massenfluss und eine hohe Streuung der Schichtdicken einzelner Proben verursachen.

- **Thermische Verformungen**

Nicht vollständig korrigierte thermische Verformungen sowie unerkannte leichte Aufschmelzungen der Beschichtung führen zu einer inkorrekten Profilmessung.

- **Zufällige Partikelagglomerationen**

Anhaftende Partikelagglomerationen teilgeschmolzener Partikel beeinflussen die lokal ermittelten Schichtdicken auf der Footprintoberfläche.

- **Untypische Eingriffsbedingungen**

Die Beschichtung der Footprints mit stationärem Spritzbrenner ist untypisch für thermische Spritzprozesse und führt zu einer punktuellen thermischen Belastung der Beschichtung und höheren Schichttemperaturen, welche in der Regel das Haftungsverhalten von Partikeln auf der Oberfläche beeinflussen.

- **Messfehler**

Trotz aller Sorgfalt bei der Vermessung der Footprintprofile bleibt die Messung der im Vergleich zur lateralen Ausdehnung sehr dünnen Schichtprofile eine Herausforderung für moderne Scanner und es verbleibt stets ein Restfehler bei der Vermessung.

In diesem Kapitel wird daher ein Ansatz zur Kalibrierung der Simulation anhand von gemessenen Schichtprofilen vorgestellt, welche auf Basis von Spritzbahnen mit nicht stationärer Spritzpistole hergestellt werden. Die grundlegende Idee besteht dabei darin, die in der Simulation verwendeten Footprintprofile so zu variieren beziehungsweise zu optimieren, dass die für ein oder mehrere Spritzbahnen auf Basis dieser variierten Footprints simulierten Schichtprofile möglichst genau mit experimentell ermittelten Profilen dieser Spritzbahnen übereinstimmen. Dass sich eine durch diese Modifikation der in der Simulation verwendeten Footprintprofile erzielte Verbesserung der Simulationsgenauigkeit nicht nur auf die zur Optimierung der Footprints verwendeten Spritzbahnen beschränkt, sondern auch auf weitere Simulationen übertragen werden kann, ist darin begründet, dass die zur Kalibrierung genutzten Bahnen praxisnähere Beschichtungsabläufe darstellen als die mit stationärem Brenner erzeugten Footprints. Insbesondere nimmt die thermische Belastung der Beschichtung aufgrund der permanenten Bewegung des Brenners deutlich ab und ein Aufschmelzen kann nahezu ausgeschlossen werden. Des Weiteren werden störende Einflüsse von Prozessfluktuationen infolge einer im Regelfall deutlich längeren Gesamtspritzdauer minimiert.

Im folgenden Abschnitt 3.8.1 wird zunächst eine Deformationsfunktion zur lokalen Variation des verwendeten Footprintprofils eingeführt. Die Bestimmung der Parameter dieser Deformationsfunktion stellt das Ziel der Profilloptimierung mittels der in Kapitel 3.8.3 aufgestellten Zielfunktion dar, welche die Abweichungen zwischen simulierten und gemessenen Querschnittsprofilen minimiert. Als Querschnittsprofile dienen dabei Profile, die mit linearen Verfahren des Brenners erzeugt wurden und für die in Kapitel 3.8.2 eine einfache Berechnungsvorschrift hergeleitet wird. In Kapitel 3.8.4 werden die Ergebnisse dieses Ansatzes vorgestellt und Kapitel 3.8.5 betrachtet den Nutzen der in Kapitel 3.6.3 vorgestellten Methodik zur Korrektur thermischer Verformungen im Hinblick auf den Simulationsfehler für die verwendeten Linearbahnprofile.

3.8.1 Variation des Footprintprofils

Die Variation des Footprints $\mathcal{F}(x, y)$ zur Kalibrierung der Simulation erfolgt durch Multiplikation mit einer zu bestimmenden Funktion $C_n(x, y)$, welche aus der bikubischen Interpolation von $n \times n$ auf einem regulären Gitter angeordneten Stützstellenwerten $c_{i,j}$ hervorgeht:

$$\mathcal{F}_d(x, y) = C_n(x, y) \cdot \mathcal{F}(x, y). \quad (3.7)$$

Zusätzlich zu dieser Deformation werden gegebenenfalls Verschiebungen δ_x , δ_y und δ_z des Profils sowie eine Skalierung s_z der Profilhöhe als Parameter in die Optimierung einbezogen, um eventuelle Fehler bei der Bestimmung des Arbeitspunktes (TCP) ($\rightarrow \delta_x, \delta_y$; siehe auch Kapitel 3.4.1) oder der Nullebene des Profils ($\rightarrow \delta_z$) sowie Abweichungen der tatsächlichen

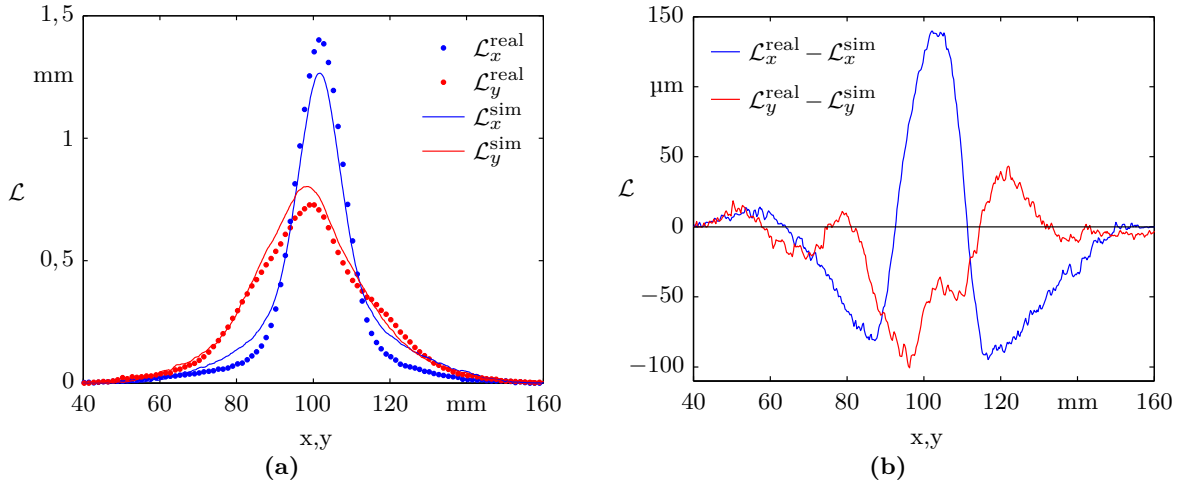


Abbildung 3.31: Verlauf der real gemessenen und der mit $\mathcal{F}_{\text{loose}}$ simulierten Linearbahnprofile \mathcal{L}_x^* und \mathcal{L}_y^* für beide Linearbahnen in x- respektive y-Richtung (a), sowie deren Differenzen (b). Die Anzahl der Überläufe beträgt $m = 26$ und die Verfahrensgeschwindigkeit $v = 200 \text{ mm/s}$.

von der programmierten Footprintspritzdauer $t_{\mathcal{F}}$ beziehungsweise der Bahngeschwindigkeit v ($\rightarrow s_z$; Gleichungen (3.10) und (3.11)) zu kompensieren. Die Simulation der Spritzbahnen im Rahmen der Kalibrierung erfolgt damit insgesamt auf Basis des transformierten und deformierten Footprints

$$\mathcal{F}_{td}(x, y) = C_n(x - \delta_x, y - \delta_y) \cdot [s_z \mathcal{F}(x - \delta_x, y - \delta_y) + \delta_z]. \quad (3.8)$$

Die n^2 Parameter $c_{i,j}$ der Funktion C_n sowie die vier Transformationsparameter stellen die $n^2 + 4$ Parameter der zu minimierenden Zielfunktion dar, welche in Gleichung (3.12) definiert wird. Initialisiert werden diese Parameter bei allen im Folgenden beschriebenen Optimierungen mit der neutralen Belegung $c_{i,j} = 1$, $\delta_x = \delta_y = \delta_z = 0$ und $s_z = 1$, welche den ursprünglichen, undeformierten Footprint $\mathcal{F}(x, y)$ wiedergibt. Zu Vergleichszwecken werden im weiteren Verlauf des Kapitels sowohl Ergebnisse optimierter Footprintprofile auf Basis sowohl von \mathcal{F}_{td} als auch von \mathcal{F}_d betrachtet.

3.8.2 Definition der Spritzbahnen

Bevor auf die genaue Formulierung der Zielfunktion eingegangen wird, wird nachstehend zunächst die Wahl der für die Optimierung verwendeten Beschichtungsbahnen diskutiert. Prinzipiell können im Rahmen der Optimierung unter Zuhilfenahme der Simulation beliebige Bahnverläufe betrachtet werden. An dieser Stelle soll jedoch durch eine sinnvolle Wahl der Bahnen der Berechnungsaufwand reduziert und ein eigenständiges, von der in Kapitel 5 vorgestellten Simulation unabhängiges Verfahren realisiert werden. Da die Berechnung der von den Bahnverläufen abgelagerten Profile für jede Auswertung der Zielfunktion erneut durchgeführt werden muss, werden an dieser Stelle zwei Linearbahnen konstanter Geschwindigkeit

und konstanten Abstands vom ebenen Substrat betrachtet, welche in x- bzw. y-Richtung im Werkzeugkoordinatensystem verlaufen. Als Spritzprofile ergeben sich somit zwei extrudierte, im Querschnitt näherungsweise gaußförmige Linearbahnprofile $\mathcal{L}_x(x, y)$ bzw. $\mathcal{L}_y(x, y)$ für die Bahnen in x- respektive y-Richtung (Abbildung 3.31a). Aufgrund der linearen Bahnverläufe mit konstanter Geschwindigkeit v ist die Trajektorie jedes Punktes (x, y) der Substratoberfläche eine gerade Linie durch den Spritzkegel des Brenners. Der simulierte Schichtauftrag an diesem Punkt entspricht damit einer Aufsummierung von Werten des verwendeten Footprintprofils $\mathcal{F}(x, y)$ ebenfalls entlang einer geraden Linie $x = vt$. Für die Spritzbahn in x-Richtung ergibt sich:

$$\mathcal{L}_x^{\text{sim}}(x, y) = \mathcal{L}_x^{\text{sim}}(y) = m \cdot \int_{-\infty}^{\infty} \frac{1}{t_{\mathcal{F}}} \mathcal{F}_{td}(vt, y) dt.$$

Darin ist m die Anzahl der Überläufe der Linearbahn. Der auf die Spritzdauer $t_{\mathcal{F}}$ normierte Footprint $\mathcal{F}_{td}(\cdot, \cdot)/t_{\mathcal{F}}$ gibt die Schichtwachstumsrate in mm/s an. Der konstante Abstand vom Substrat entspricht dem Abstand bei der Erzeugung der Footprints (110 mm). Mit dem Definitionsbereich $(0, l_{\mathcal{F}}) \times (0, l_{\mathcal{F}})$ des quadratisch angenommenen Footprints sowie der Substitution $dt = dx/v$ ergibt sich

$$\mathcal{L}_x^{\text{sim}}(y) = \frac{m}{vt_{\mathcal{F}}} \cdot \int_0^{l_{\mathcal{F}}} \mathcal{F}_{td}(x, y) dx \quad (3.9)$$

beziehungsweise in diskretisierter Form für einen Footprint aus $N \times N$ Höhenwerten sowie $\Delta x = \Delta y = l_{\mathcal{F}}/(N - 1)$

$$\begin{aligned} \mathcal{L}_x^{\text{sim}}(j\Delta y) &= \frac{m}{vt_{\mathcal{F}}} \cdot \sum_{i=0}^{N-1} \mathcal{F}_{td}(i\Delta x, j\Delta y) \Delta x \\ &= \frac{m}{vt_{\mathcal{F}}} \frac{l_{\mathcal{F}}}{N-1} \cdot \sum_{i=0}^{N-1} \mathcal{F}_{td}(i\Delta x, j\Delta y) \end{aligned} \quad (3.10)$$

für $j = 0, \dots, N - 1$. Analog ergibt sich für die Linearbahn in y-Richtung und $i = 0, \dots, N - 1$

$$\mathcal{L}_y^{\text{sim}}(i\Delta x) = \frac{m}{vt_{\mathcal{F}}} \frac{l_{\mathcal{F}}}{N-1} \cdot \sum_{j=0}^{N-1} \mathcal{F}_{td}(i\Delta x, j\Delta y). \quad (3.11)$$

Die footprintgebundene Simulation der Linearbahnprofile vereinfacht sich durch diese Wahl der Beschichtungsbahnen also wesentlich, da lediglich skalierte Zeilen- und Spaltensummen der diskreten $N \times N$ -Footprintmatrix berechnet werden müssen. Ebenso vorteilhaft erweist sich die Wahl dieser Bahnen für die Messung der Querschnittsprofile $\mathcal{L}_x^{\text{real}}(y)$ bzw. $\mathcal{L}_y^{\text{real}}(x)$ an den realen Proben, da diese, aufgrund der extrudierten Form der Profile, als rauscharme Durchschnittsprofile entlang der x- respektive y-Achse ermittelt werden können.

Hansbo et al. zogen bereits eine ähnliche Konfiguration aus zwei sich orthogonal schneidenden Linearbahnen zur Bestimmung eines Footprints als Gitter von $N \times N$ Datenpunkten in Betracht [66]. Da jedoch die obigen Gleichungen (3.10) und (3.11) nur $2N$ lineare Gleichungen für N^2 Unbekannte des Footprintprofils liefern und sie die Durchführung von zusätzlichen Versuchen zum Erhalt weiterer, unabhängiger Gleichungen als zu aufwendig einstufen, beschränkten sich Hansbo et al. auf die Beschichtung rotierender Objekte, für die die Messung

einer einzigen Bahn ausreichend ist. Die eindeutige Lösbarkeit dieser Gleichungen ist jedoch für die Optimierung des Footprintprofils nicht erforderlich.

3.8.3 Zielfunktion

Auf Basis der simulierten Linearbahnprofile sowie der gemessenen Profile kann nun eine Zielfunktion aufgestellt werden, welche neben der Abweichung der Profile der Bahnen in x- und y-Richtung auch einen Glattheitsterm f_{smooth} für die Deformationsfunktion $C_n(x, y)$ sowie einen Term f_{similar} zur Beschränkung der Abweichung des resultierenden optimierten Footprints vom gemessenen Ausgangsfootprint beinhaltet:

$$f_{\text{opt}}(c_{00}, \dots, c_{nn}, \delta_x, \delta_y, \delta_z, s_z) = (1 - \alpha)(f_x + f_y + \lambda f_{\text{smooth}}) + \alpha f_{\text{similar}}. \quad (3.12)$$

Darin sind

$$f_x = \sum_{j=0}^{N-1} \left(\mathcal{L}_x^{\text{sim}}(j \cdot \Delta y) - \mathcal{L}_x^{\text{real}}(j \cdot \Delta y) \right)^2 \quad (3.13)$$

und

$$f_y = \sum_{i=0}^{N-1} \left(\mathcal{L}_y^{\text{sim}}(i \cdot \Delta x) - \mathcal{L}_y^{\text{real}}(i \cdot \Delta x) \right)^2 \quad (3.14)$$

als quadratische Abweichung des Simulationsergebnisses vom Messergebnis definiert. Der Glattheitsterm, welcher sicherstellt, dass die Deformationsfunktion C_n keine zu starken Schwankungen aufweist, wird als Durchschnitt aller lokalen quadratischen Differenzen benachbarter Stützstellen $c_{i,j}$ realisiert:

$$f_{\text{smooth}} = \frac{1}{2N(N-1)} \sum_{i=1}^N \sum_{j=1}^{N-1} \left((c_{j+1,i} - c_{j,i})^2 + (c_{i,j+1} - c_{i,j})^2 \right).$$

Über den heuristisch gewählten Parameter λ in Gleichung (3.12) kann die Größenordnung von $\lambda f_{\text{smooth}}$ in Relation zu der von f_x und f_y eingestellt werden. Ferner kann über den Parameter α die Gewichtung der Ähnlichkeitsforderung

$$f_{\text{similar}} = \frac{1}{N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (\mathcal{F}_{td}(i \cdot \Delta x, j \cdot \Delta y) - \mathcal{F}(i \cdot \Delta x, j \cdot \Delta y))^2 \quad (3.15)$$

gesteuert werden.

Die beschriebene Zielfunktion wurde in der Programmierumgebung Matlab¹⁷ in beiden Varianten für die Footprintmodifikation mit N^2 bzw. $N^2 + 4$ Parametern implementiert (Gleichungen (3.7) und (3.8)). Als Optimierungsverfahren wurde der zur Lösung nichtlinearer Probleme mit Nebenbedingungen entworfene und in Matlab implementierte Algorithmus *Opt-Quest/NLP* von Ugray et al. [162] verwendet. Zur Suche eines globalen Optimums werden

¹⁷<http://www.mathworks.com>

dabei zusätzlich zu dem gegebenen Startpunkt mehrere Startpunkte mittels eines Scatter-Search-Ansatzes erzeugt [68]. Als Nebenbedingungen werden die Optimierungsparameter dabei wie folgt beschränkt, um negative $c_{i,j}$ sowie dem Betrag nach zu große, unplausible Transformationsparameter zu vermeiden:

$$-50 \text{ mm} \leq \delta_x, \delta_y \leq 50 \text{ mm}, \quad -5 \text{ mm} \leq \delta_z \leq 5 \text{ mm}, \quad 0.5 \leq s_z \leq 10, \quad 0 \leq c_{i,j}.$$

3.8.4 Ergebnisse

Um die Güte der in Kapitel 3.6.4 beschriebenen unoptimierten Footprints \mathcal{F}_{loose} und \mathcal{F}_{tight} zu beurteilen und einen Vergleich mit den optimierten Ergebnissen zu ermöglichen, wurden für diese Footprints zunächst die simulierten Profile $\mathcal{L}_*^{\text{sim}}$ berechnet und mit den gemessenen Profilen verglichen. Abbildung 3.31a zeigt für \mathcal{F}_{loose} die Verläufe von $\mathcal{L}_*^{\text{sim}}$ und $\mathcal{L}_*^{\text{real}}$ beider Linearbahnen in x- und y-Richtung. Anhand der in Abbildung 3.31b visualisierten Differenzen zwischen den Simulations- und den Messergebnissen, welche auch für den Footprint \mathcal{F}_{tight} eine ähnliche Charakteristik aufweisen, ist zu erkennen, dass das simulierte Profil $\mathcal{L}_x^{\text{sim}}$ tendenziell zu flach und an den Flanken zu breit ist. Das Profil $\mathcal{L}_y^{\text{sim}}$ in orthogonaler Richtung ist zu hoch und an den Flanken minimal zu schmal. Um diese Differenzen minimieren zu können, muss der Optimierungsalgorithmus eine Variation des initialen Footprintprofils bestimmen, welche das Integral über den Footprint in x-Richtung in der Mitte anhebt und gleichzeitig das Integral in y-Richtung in der Mitte absenkt.

Es wurden mehrere Optimierungsläufe durchgeführt, wobei verschiedene initiale Footprints (\mathcal{F}_{loose} und \mathcal{F}_{tight}), Deformationsfunktionen $C_n(x,y)$ mit unterschiedlich vielen Stützstellen ($n^2 = 3^2, 5^2, 7^2, 9^2, 15^2, 25^2, 51^2$) sowie beide Formulierungen zur Variation des Ausgangsfootprints (Gleichungen (3.7) und (3.8)) verwendet wurden. Die resultierenden Profile werden im Folgenden mit $\mathcal{F}_{loose,d}$, $\mathcal{F}_{loose,td}$, $\mathcal{F}_{tight,d}$ und $\mathcal{F}_{tight,td}$ bezeichnet. Dabei geben die zusätzlichen Indizes \bullet_d und \bullet_{td} die Art der Variation des Ausgangsfootprints an: ausschließlich durch Deformation nach Gleichung (3.7) bzw. durch Deformation und Transformation gemäß Gleichung (3.8). Insgesamt wurden hierfür $2 \times 2 \times 7 = 28$ Optimierungsläufe mit Parameteranzahlen zwischen 9 ($\mathcal{F}_{*,d}$, $C_3(x,y)$) und 2.605 ($\mathcal{F}_{*,td}$, $C_{51}(x,y)$) durchgeführt. Um möglichst gute Zielfunktionswerte, insbesondere für f_x und f_y , zu erzielen, wurden zunächst beliebig große Deformationen erlaubt und die Ähnlichkeitsforderung f_{similar} durch Setzen von $\alpha = 0$ ausgeklammert. Der Gewichtungsfaktor λ für f_{smooth} wurde vor der Durchführung der Optimierungsläufe so eingestellt, dass $\lambda f_{\text{smooth}}$ die gleiche Größenordnung aufweist wie f_x und f_y . Eine Vernachlässigung dieses Terms ($\lambda = 0$) kann unerwünschte lokale Schwankungen in den Stützstellen $c_{i,j}$ zur Folge haben.

Die finalen Zielfunktionsterme $f_x + f_y$ der Ergebnisse sind zusammen mit denen der nicht optimierten Footprints in Abbildung 3.32 dargestellt. Daraus geht zunächst hervor, dass das erzielte Ergebnis erwartungsgemäß mit zunehmender Auflösung der Deformationsfunktion deutlich verbessert wird. Ebenfalls erwartungsgemäß erzielen die Optimierungsläufe mit zusätzlichen Transformationsparametern aufgrund der zusätzlichen Freiheitsgrade bessere Ergebnisse als die rein deformationsbasierten. Obwohl der initiale Zielfunktionswert des unopti-

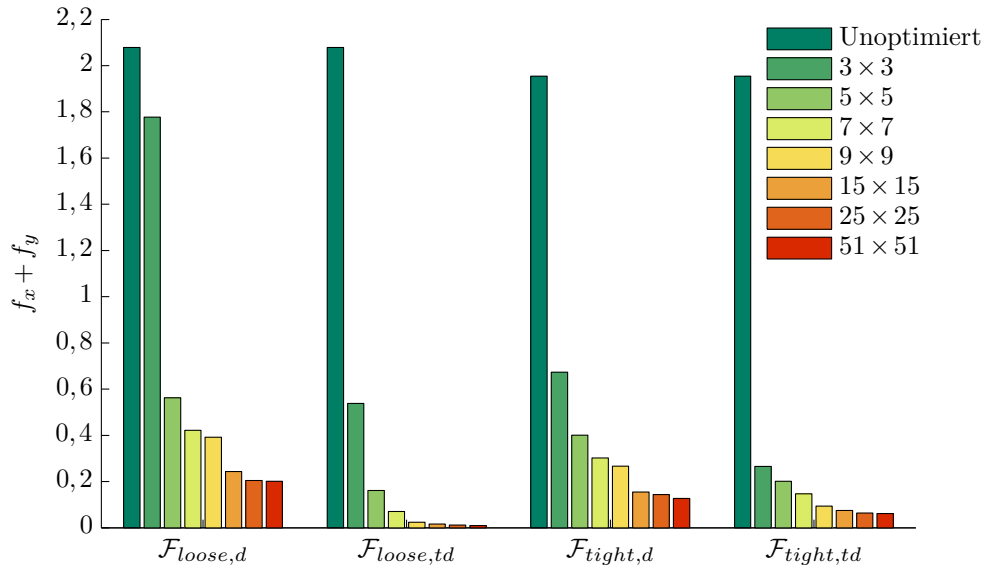


Abbildung 3.32: Ergebnisse der Optimierung für die beiden Footprints \mathcal{F}_{loose} und \mathcal{F}_{tight} mit (Index ‘*,td’) und ohne (Index ‘*,d’) einer Optimierung der vier Transformationsparameter δ_x , δ_y , δ_z und s_z für verschiedene Auflösungen der Deformationsfunktion von $n = 3$ bis $n = 51$. Als Vergleichswerte sind ebenfalls die Werte $f_x + f_y$ der unoptimierten Footprints angegeben.

mierten Footprints \mathcal{F}_{loose} geringfügig größer ist als der von \mathcal{F}_{tight} , ergibt sich das beste Ergebnis $f_x + f_y = 9,81 \cdot 10^{-3}$ für $\mathcal{F}_{loose,td}$ bei maximal aufgelöster Deformationsfunktion ($n = 51$). Abbildung 3.33 zeigt für drei Konfigurationen die berechneten Deformationsfunktionen sowie die resultierenden simulierten Profile $\mathcal{L}_*^{\text{sim}}$ im Vergleich mit den real gemessenen Profilen. Besonders bei der ersten Konfiguration mit $\mathcal{F}_{tight,td}$ und $n = 7$ wird deutlich, in welchen Bereichen die Profilhöhe des ursprünglichen Footprints vergrößert oder verkleinert wird, um $\mathcal{L}_x^{\text{sim}}$ in der Mitte anzuheben und gleichzeitig $\mathcal{L}_y^{\text{sim}}$ in der Mitte abzusenken. Während in diesem Fall bereits eine Annäherung der resultierenden simulierten Profile an die gemessenen sichtbar ist (Abbildung 3.33b), liefern die unteren Footprints $\mathcal{F}_{tight,td}$ und $\mathcal{F}_{loose,td}$ mit $n = 51$ im Rahmen der Simulation nahezu perfekte Profile (Abbildung 3.33d und 3.33f). Der aus dieser Optimierung resultierende Footprint $\mathcal{F}_{loose,td}$ ist im Vergleich zum ursprünglichen Footprint noch etwas langgestreckter, also breiter in x-Richtung und weniger ausgedehnt in y-Richtung (Abbildung 3.34).

Die alleinige Minimierung der Abweichungen zwischen den simulierten und den gemessenen Querschnittsprofilen ist jedoch für die Verwendung des resultierenden Footprints für Bahnen, die nicht waagrecht oder senkrecht verlaufen, nicht aussagekräftig: Wird die Optimierung nicht mit einem gemessenen Footprintprofil wie zum Beispiel \mathcal{F}_{loose} oder \mathcal{F}_{tight} gestartet, sondern mit einem konstanten Footprint $\mathcal{F}(x,y) = 1$, so ergibt sich im Fall von $n = 51$ die in Abbildung 3.35 dargestellte Lösung. Diese erfüllt zwar Gleichungen (3.13) und (3.14), stellt jedoch offensichtlich kein realistisches Footprintprofil dar und ist nicht zur Simulation von Bahnen in beliebigen Bewegungsrichtungen geeignet. Dass die zuvor beschriebenen, in Ab-

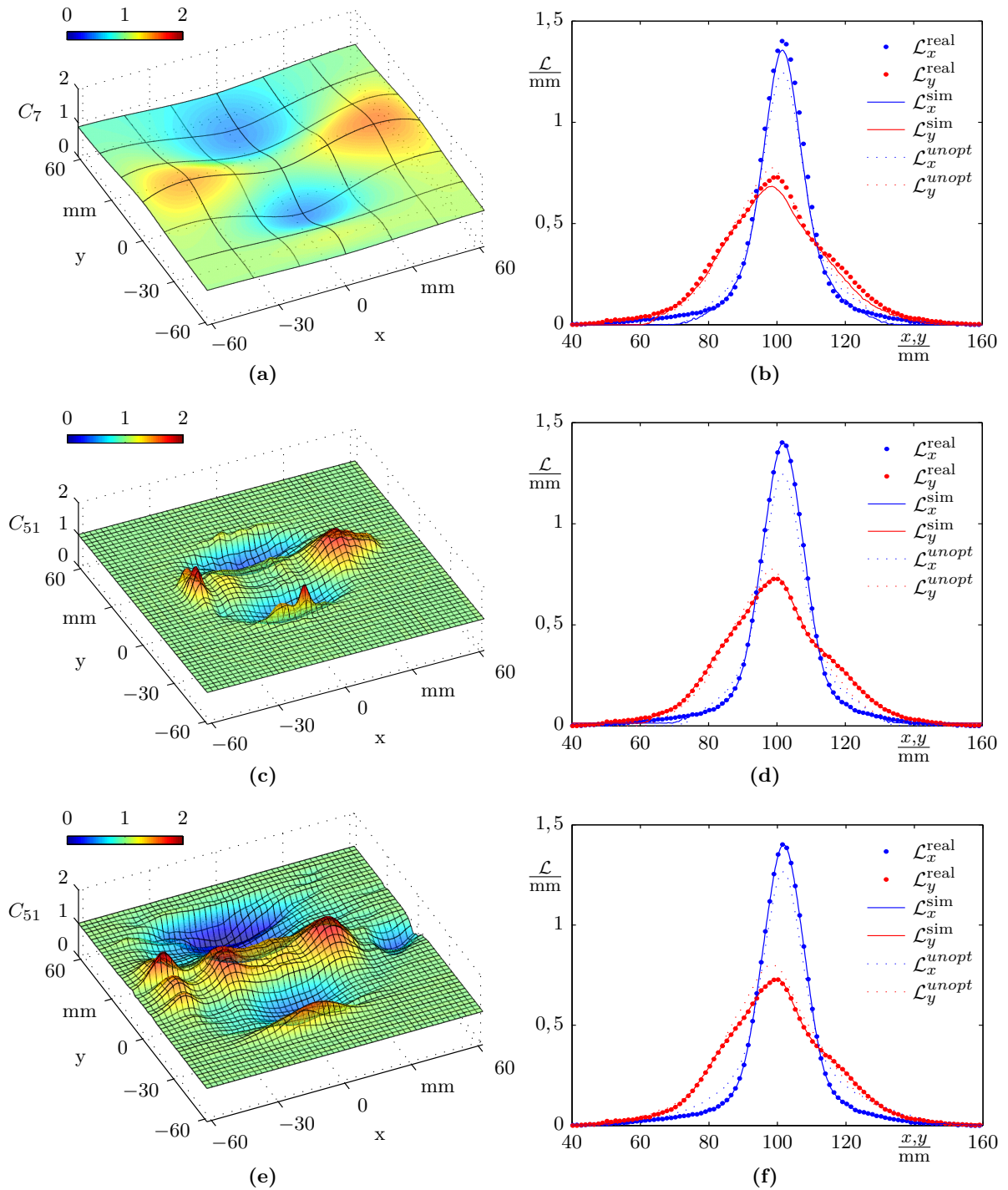


Abbildung 3.33: Aus der Optimierung resultierende Deformationsfunktionen (links) sowie ein Vergleich der daraus hervorgehenden simulierten und der real gemessenen Profile in x- und y-Richtung (rechts) für drei verschiedene Optimierungsläufe. Um die durch die Optimierung erzielte Verbesserung der Linearbahnprofile beurteilen zu können, sind zusätzlich die simulierten Profile auf Basis der unoptimierten Footprints $\mathcal{L}^{\text{unopt}}$ eingezeichnet. (a),(b): $\mathcal{F}_{\text{tight},td}$ mit $n = 7$. (c),(d): $\mathcal{F}_{\text{tight},td}$ mit $n = 51$. (e),(f): $\mathcal{F}_{\text{loose},td}$ mit $n = 51$.

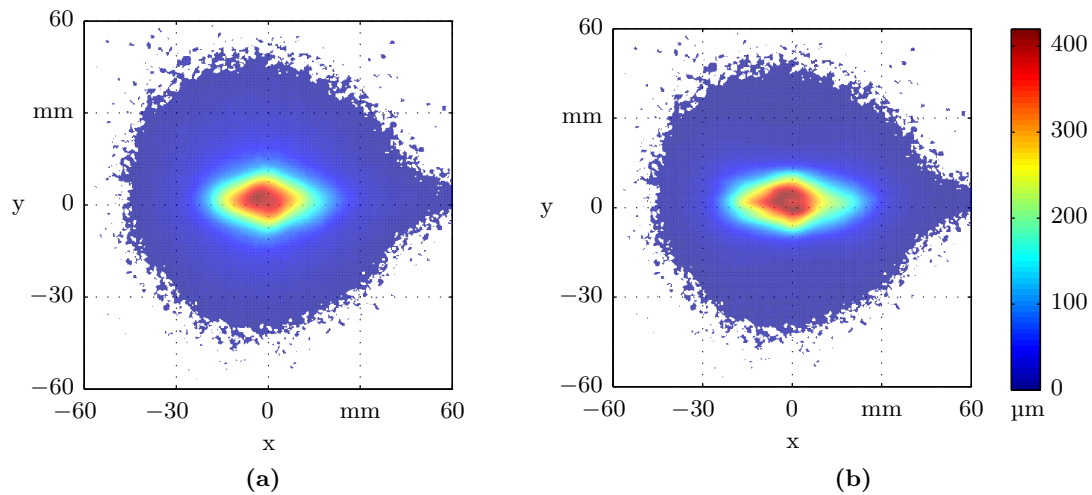


Abbildung 3.34: (a) Originaler Footprint \mathcal{F}_{loose} und (b) der aus der Optimierung resultierende Footprint $\mathcal{F}_{loose,td}$ mit $n = 51$. Letzterer ist in x-Richtung gestreckter und in y-Richtung schmäler als \mathcal{F}_{loose} und erreicht von allen Optimierungsläufen das beste Ergebnis (vgl. Abbildung 3.32).

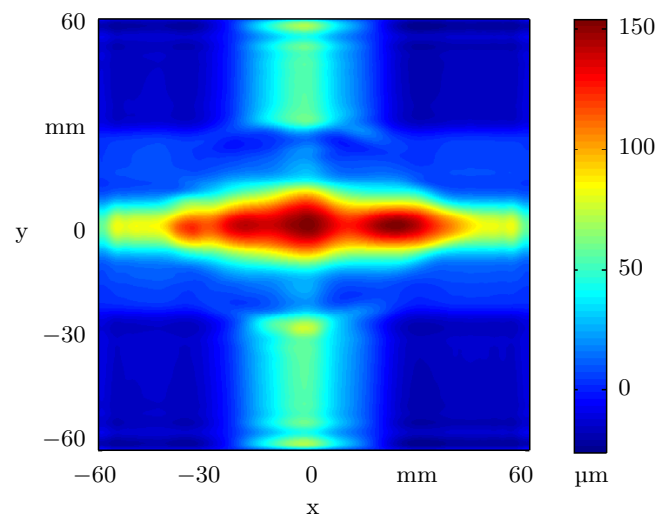


Abbildung 3.35: Footprint als Resultat einer Optimierung mit $n = 51$ ausgehend von einem konstantem Footprint $\mathcal{F}(x, y) = 1$ als Startpunkt. Es ist deutlich erkennbar, dass der Footprint nur auf einer Optimierung parallel zu seinen Achsen basiert. In diese Richtungen integriert erfüllt er die Zielfunktion ebenso genau wie die aus einer Optimierung von \mathcal{F}_{loose} oder \mathcal{F}_{tight} resultierenden Footprints. Zur Berechnung von Beschichtungen auf Basis beliebiger Roboterbahnen ist er jedoch ungeeignet. Das Akquirieren eines realen Footprints als Startpunkt für die Optimierung ist also weiterhin notwendig.

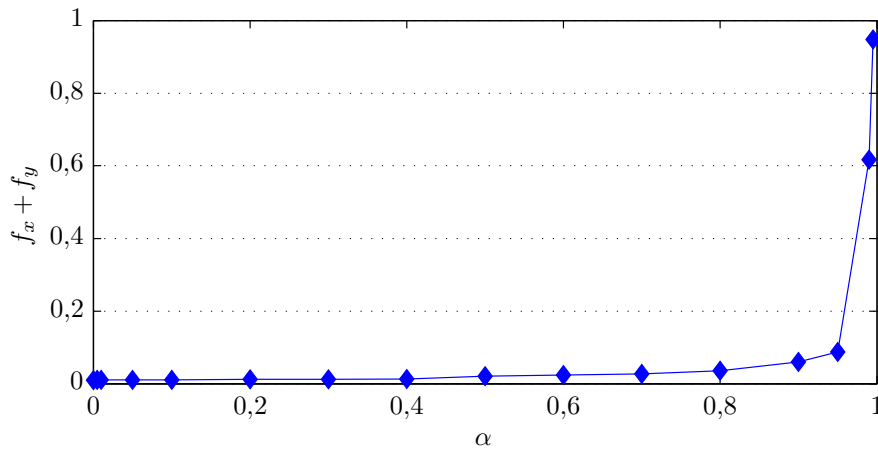


Abbildung 3.36: Approximationsfehler $f_x + f_y$ der Linearbahnen durch die optimierten Footprints in Abhängigkeit der Gewichtung α des Ähnlichkeitsterms (Gleichung (3.15)). Mit zunehmender Gewichtung steigt der Approximationsfehler zunächst leicht und ab ca. $\alpha = 0,95$ deutlich an.

bildung 3.33 dargestellten Optimierungsergebnisse realen Profilen stark ähneln, ist zunächst lediglich auf die Wahl eines real gemessenen Footprintprofils als Startpunkt für die Optimierung zurückzuführen. Um diese Ähnlichkeitsbedingung bereits während der Optimierung zu berücksichtigen, kann in der Zielfunktion die in Gleichung (3.15) formulierte Ähnlichkeitsforderung f_{similar} verwendet werden. Abbildung 3.37 zeigt hierzu die für Gewichtungen $\alpha = 0,01$, $\alpha = 0,5$ sowie $\alpha = 0,99$ aus der Optimierung resultierenden Deformationsfunktionen $C_{51}(x, y)$ für $\mathcal{F}_{\text{loose,td}}$. Es ist erkennbar, dass sich $C_{51}(x, y)$ mit zunehmendem α , also höher gewichtetem Term f_{similar} , der neutralen Gewichtung $C(x, y) = 1$ annähert. Diese Beschränkung der Deformation führt für steigendes α zu einem exponentiellen Anstieg von $f_x + f_y$, also einer Verschlechterung der Approximation der gemessenen Linearbahnprofile (Abbildung 3.36). Für den Bereich bis ca. $\alpha = 0,95$ ist diese Verschlechterung jedoch nur minimal und die resultierenden Werte für $f_x + f_y$ liegen noch immer deutlich (\approx Faktor 23) unter dem Wert des unoptimierten Footprints. Da die Deformationsfunktionen in diesem Bereich bereits deutlich schwächer ausgeprägt sind, erscheint es sinnvoll, diese sichtbar geringere Deformation der nur geringfügig besseren Approximation der Querschnitte bei $\alpha = 0$ vorzuziehen.

Insgesamt kann festgehalten werden, dass die vorgestellte Optimierungsmethode es ermöglicht, auf Basis eines initialen Footprints einen modifizierten Footprint zu berechnen, welcher die gegebenen gemessenen Schichtprofile zweier orthogonaler Linearbahnen in der Größenordnung der Auflösung n des Stützstellengitters in der Simulation nahezu exakt abbildet. Ein auf 51×51 Stützstellen basierendes Gitter reicht dabei für die maximal $130 \times 130 \text{ mm}^2$ großen Footprintprofile vollkommen aus, da sich daraus bereits ein Stützstellenabstand von weniger als 2,5 mm ergibt und eine wesentliche Erhöhung der Auflösung lediglich das in den Messungen enthaltene Rauschen approximieren würde.

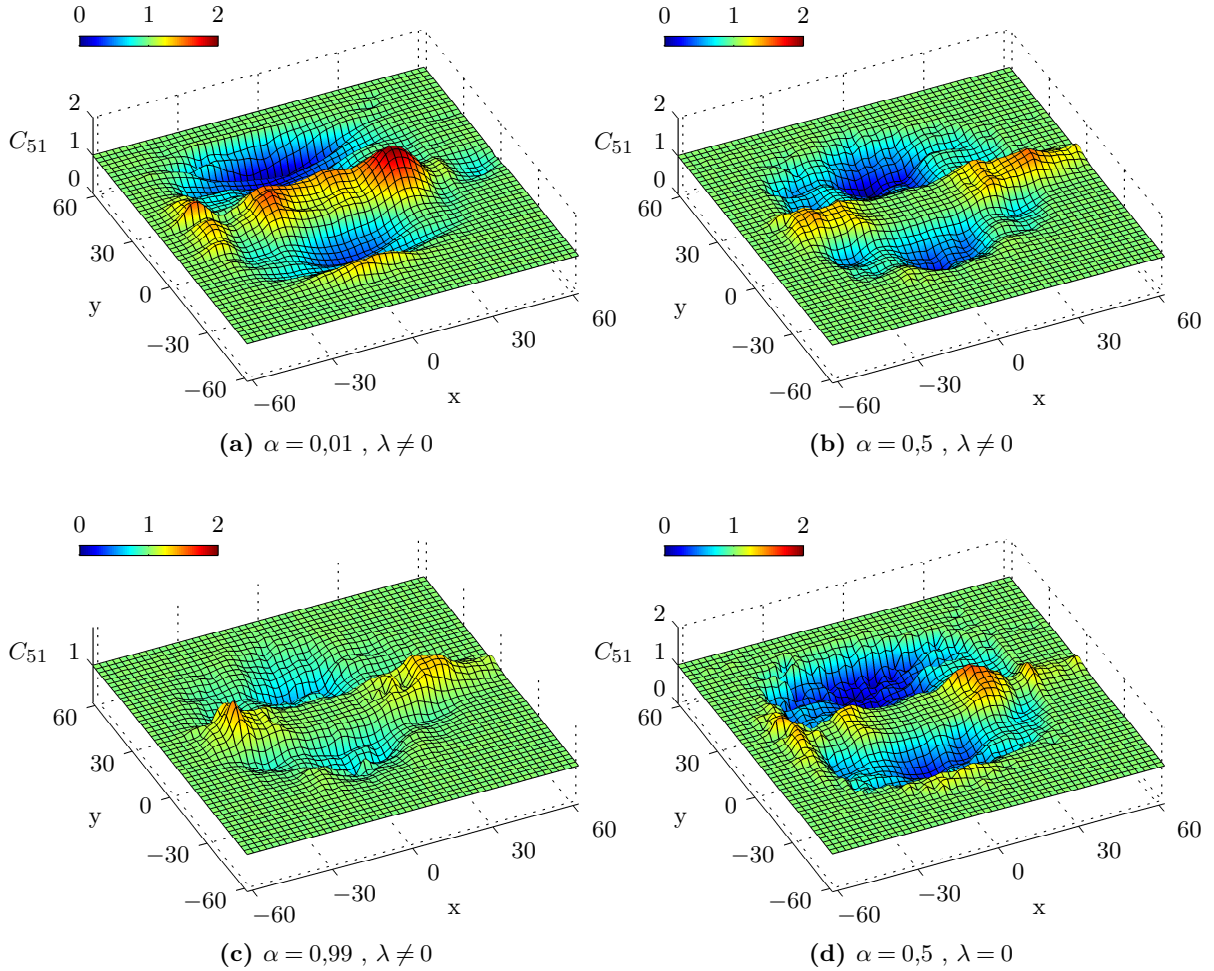


Abbildung 3.37: Auf Basis des Footprints $\mathcal{F}_{\text{loose,td}}$ berechnete Deformationsfunktionen $C_{51}(x, y)$ für verschiedene Gewichtungen α des Ähnlichkeitsterms aus Gleichung (3.15). (a) - (c): Mit zunehmender Gewichtung der Ähnlichkeitsforderung zum Ausgangsfootprint $\mathcal{F}_{\text{loose}}$ wird die Deformationsfunktion flacher. (d): Das Vernachlässigen des Glattheitsterms ($\lambda = 0$) hat im Vergleich mit (b) etwas höhere lokale Schwankungen der Deformationsfunktion zur Folge.

3.8.5 Ergebnisse zur Effektivität der Korrektur thermischer Verformungen

Abgesehen von der Ermittlung eines optimalen Footprints kann auf Basis der aufgestellten Zielfunktion und der gemessenen Linearbahnprofile untersucht werden, inwieweit die in Kapitel 3.6.3 beschriebenen Methoden für den Ausgleich thermischer Verformungen die Genauigkeit der simulierten Linearbahnprofile verbessern. Zu diesem Zweck wurden simulierte Profile für beide Linearbahnrichtungen für alle acht gefertigten Einzelfootprints berechnet und mit den gemessenen Profilen $\mathcal{L}_*^{\text{real}}$ verglichen. Zur Berechnung der Abweichungen wurde hier ebenfalls die Zielfunktion f_{opt} aus Gleichung (3.12) verwendet, wobei jedoch zunächst $C_n(x, y) = 1$ sowie $\delta_x = \delta_y = \delta_z = 0$ und $s_z = 1$ gesetzt wurden, um die durch den Ausgleich

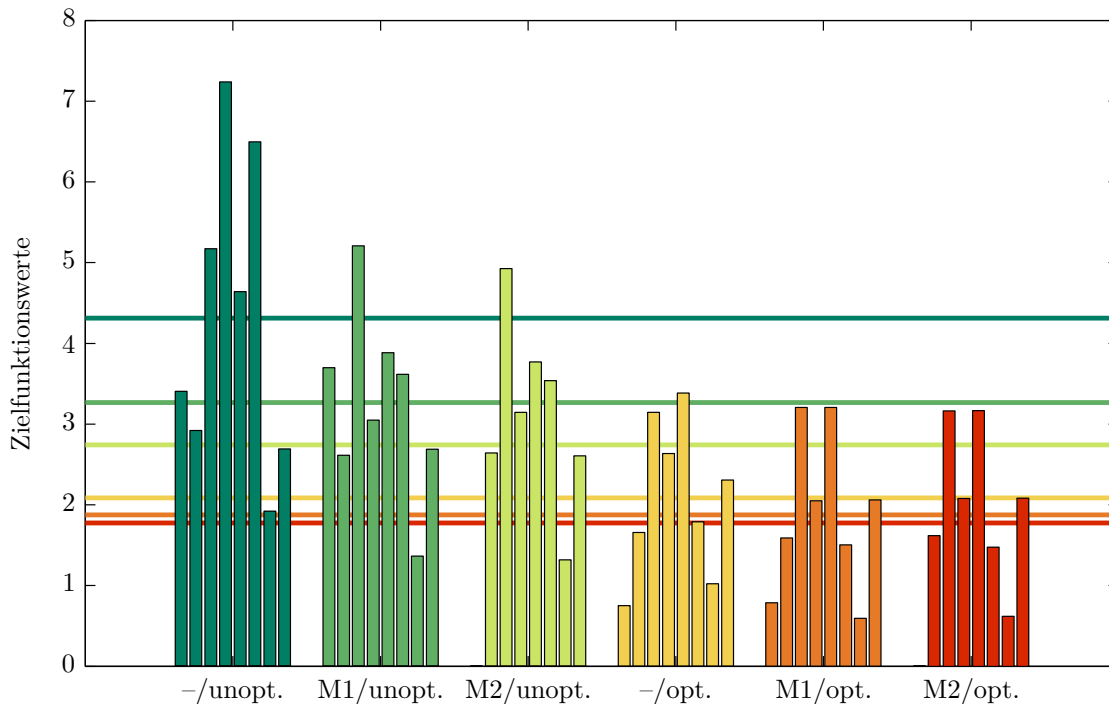


Abbildung 3.38: Zielfunktionswerte für alle acht Footprints mit (rechts) und ohne (links) Optimierung der Translation δ_x , δ_y , δ_z und der Skalierung s_z für jeweils drei unterschiedlich kompen-sierte thermische Verformungen. Dabei steht „-“ für „keine Kompensation“ und „M1“ bzw. „M2“ stehen für die Kompensationen nach der ersten bzw. zweiten in Kapitel 3.6.3 vorgestellten Metho-de. Die waagerechten Linien geben die durchschnittlichen Zielfunktionswerte der gleichfarbigen Footprints an.

der thermischen Verformungen erzielten Veränderungen direkt und ohne eine weitere Opti-mierung beurteilen zu können. Für die acht Footprints wurden zunächst jeweils drei Linear-bahnprofile \mathcal{L}^{sim} berechnet, welche auf verschiedenen Versionen der Footprintprofile basieren: ohne Ausgleich der thermischen Verformungen sowie mit Ausgleich nach Gleichung (3.1) (Methode 1) beziehungsweise nach Gleichung (3.3) (Methode 2). Anschließend wurde zusätzlich betrachtet, inwieweit eine Optimierung ausschließlich der vier Transformationsparameter δ_x , δ_y , δ_z und s_z diese Ergebnisse verbessert. Die berechneten Abweichungen für alle genann-ten Fälle sind in Abbildung 3.38 zusammengefasst. Die starke Variation der Ergebnisse der einzelnen Footprints lässt auf eine hohe Prozessvariation schließen. Innerhalb der Gruppe der nicht optimierten Footprints zeigt sich, dass die Anwendung beider Methoden zum Aus-gleich der thermischen Verformungen die Abweichungen zwischen Simulation und Experiment verringert haben, wobei die zweite Methode etwas bessere Ergebnisse erzielt. Nach der Opti-mierung weisen die Footprints mit und ohne ausgeglichener Substratverformung ähnlichere, insgesamt leicht verbesserte Abweichungswerte auf, welche insgesamt jedoch nicht an die Wer-te der bei der Optimierung zusätzlich deformierten Footprints heranreichen. Zumindest für die betrachteten Footprintproben und Linearbahnen kann daher gefolgert werden, dass sich die Korrektur thermischer Verformungen auch bei der durch die dicken Substrate bereits stark

reduzierten Verformung positiv auswirkt, wenn keine Optimierung durchgeführt wird. Nach der Optimierung der vier genannten Transformationsparameter verringert sich dieser Vorteil und bei der zusätzlichen Verwendung einer hochaufgelösten Deformationsfunktion C_n ist der initial verwendete Footprint im Wesentlichen nur noch als ein Startwert für die Optimierung der Werte $c_{i,j}$ anzusehen. Der aus der Optimierung aller Parameter hervorgehende optimale Footprint ist also so stark modifiziert, dass Unterschiede in der Initialisierung zwar noch rechnerisch bemerkbar, praktisch jedoch weitestgehend irrelevant sind. Auf der anderen Seite konvergiert der Optimierungsalgorithmus ohne einen Footprint als Startwert nicht in einer sinnvollen Lösung, wie in Abbildung 3.35 zu sehen ist. Der dargestellte Footprint weist zwar, wie in den Gleichungen (3.13) und (3.14) gefordert, geringe Abweichungen bei der Simulation der Linearbahnen in x- und y-Richtung auf, für dreidimensionale Beschichtungen ist er jedoch ungeeignet, da das Profil deutlich von gemessenen Footprints abweicht. Um nach der Optimierung einen plausibles Footprintprofil zu erhalten, welches nicht nur für zwei sondern beliebige Bahnrichtungen geeignet ist, ist daher nach wie vor eine Footprintmessung als Ausgangspunkt erforderlich.

3.9 Funktionsbasierte Footprintrepräsentation

Die in den vorherigen Kapiteln diskutierten Footprintprofile basieren auf einer numerischen Repräsentation (Höhenfeld) der nachbearbeiteten Digitalisierdaten, welche sich ideal für den Einsatz in der entwickelten Simulation als Textur eignet. Klassische Simulationsansätze verwenden jedoch in der Regel auf Digitalisierdaten gefittete Funktionen zur Spezifikation des Massenflusses. Gut geeignet sind hierfür ein- oder zweidimensionale Gaußverteilungen, welche in den letzten 20 Jahren den Regelfall für die Modellierung des Massenflusses darstellten [3, 26, 46, 54, 63, 69, 116, 156, 159]. Die Eignung von Gaußverteilungen ist mittels des Zentralen Grenzwertsatzes bezogen auf die zufälligen Partikelflugbahnen zu begründen. Dieser besagt, dass die Summe einer großen Anzahl unabhängiger, identisch verteilter Zufallszahlen, in diesem Fall die Partikelflugrichtungen, näherungsweise normalverteilt ist [16, 159]. Sadovoy et al. erweitern diese Modellierung auf die additive Überlagerung mehrerer Gaußverteilungen zur Modellierung bi- bzw. multimodaler Footprints, welche aus der Verwendung von Spritzbrennern mit mehreren Partikelinjektoren resultieren[122].

Gegenüber den direkt auf Digitalisierdaten basierenden beziehungsweise den aus der in Kapitel 3.8 beschriebenen Optimierung resultierenden Footprints haben funktionale Beschreibungen den Nachteil, dass sie den Massenfluss unter Umständen nur unzureichend abbilden. Dies ist offensichtlich der Fall, wenn nur eine einzelne Gaußfunktion als Approximation für bi- oder multimodale Verteilungen verwendet wird. Schwerer auszumachen sind Approximationsfehler jedoch, falls die tatsächliche Verteilung nur augenscheinlich unimodal ist, in Wirklichkeit aber aus zwei oder mehr dicht überlappenden unimodalen Verteilungen besteht oder andere Besonderheiten aufweist, welche beispielsweise dem Aufbau der Spritzpistole, der Düse oder daraus resultierenden Turbulenzen geschuldet sind [54]. Ein Nachteil der direkten numerischen

Repräsentation ist hingegen die Problematik, dass einzelne Partikelagglomerationen, Substrataufwerfungen durch das Sandstrahlen oder auch Digitalisierungsfehler insbesondere in den Randbereichen des Footprintsubstrates nicht repräsentativ und schwer korrigierbar sind. Das Profil sinkt hier nicht monoton auf Null ab, sondern pendelt sich auf einem Rauschniveau ein, welches häufig auch vereinzelt, lokal beschränkte Unregelmäßigkeiten enthält. Nach dem Abschneiden des Rauschniveaus unterhalb eines festgelegten Höhenwertes, wie in Kapitel 3.6.4 beschrieben, verbleibt ein „ausgefranztes“ Profil (vergleiche Abbildung 3.26), welches selbst bei einem idealen Scanprozess auftreten würde, da stets eine Übergangszone existiert, in der statistisch bedingt nur vereinzelt Partikel auf dem Substrat auftreten. Andererseits verursacht ein Abschneiden oder Ausblenden des gemessenen Profils außerhalb eines bestimmten Fensters ein mehr oder weniger abruptes Absinken des Profils auf Null und damit unter Umständen sichtbare Artefakte in Form von interferierenden Unstetigkeiten in den Ergebnissen der Beschichtungssimulation.

Demgegenüber konvergiert eine funktionale Repräsentation auf Basis ein oder mehrerer Gaußfunktionen mit zunehmendem Abstand vom Zentrum des Footprints gegen Null und vereinzelt Fehlstellen durch Agglomerationen oder Digitalisierungsfehler werden durch den Fittingprozess effektiv gefiltert. Gemäß des Zentralen Grenzwertsatzes stellt das Modell dabei an jeder Stelle den asymptotischen Grenzwert der Schichtdicke dar und kann somit auch Bruchteile der quasi-diskreten Einzelpartikeldicken wiedergeben. Ein weiterer Vorteil einer funktionalen Beschreibung stellt ihre Kompaktheit und die somit einfachere Übertragbarkeit auf andere Simulationen dar.

Funktionsbasierte Footprintprofile wurden im Rahmen dieser Arbeit für Footprints einer zweiten Messreihe (vergleiche Kapitel 3.2) ermittelt, welche in vier verschiedenen Spritzabständen von 55–220 mm erzeugt wurden. Abbildung 3.39 zeigt die vier durchschnittlichen und mittels Methode 1 aus Kapitel 3.6.3 korrigierten Footprintprofile. Für diese Profile wurden funktionsbasierte Repräsentationen auf Basis des im folgenden Kapitel 3.9.1 beschriebenen Modells ermittelt und auf ihre Eignung überprüft (Kapitel 3.9.2). Ausgehend von den gefundenen Repräsentationen wird anschließend in Kapitel 3.9.3 die dreidimensionale Geometrie des Spritzstrahls analysiert, um für die Simulation relevante Kenndaten wie Öffnungswinkel oder Neigung des Strahls zu gewinnen.

3.9.1 Modellfunktionen

Wie bereits erwähnt, eignen sich bivariate Gaußfunktionen zur Approximation des Massenflusses und damit der Footprintprofile. Betrachtet wurden im Rahmen dieser Arbeit Modelle aus einer Überlagerung von bis zu $n = 20$ Gaußfunktionen der Form

$$\mathcal{G}_n(x, y) = \sum_{i=1}^n a_i \cdot e^{-\left(\frac{(x-x_i)^2}{2\sigma_i^2} + \frac{(y-y_i)^2}{2\tau_i^2}\right)}.$$

Dabei bezeichnen a_i die positiven Amplituden, (x_i, y_i) die Mittelpunkte und σ_i und τ_i die Standardabweichungen der Einzelfunktionen. Während die Verwendung einer zweiten Gauß-

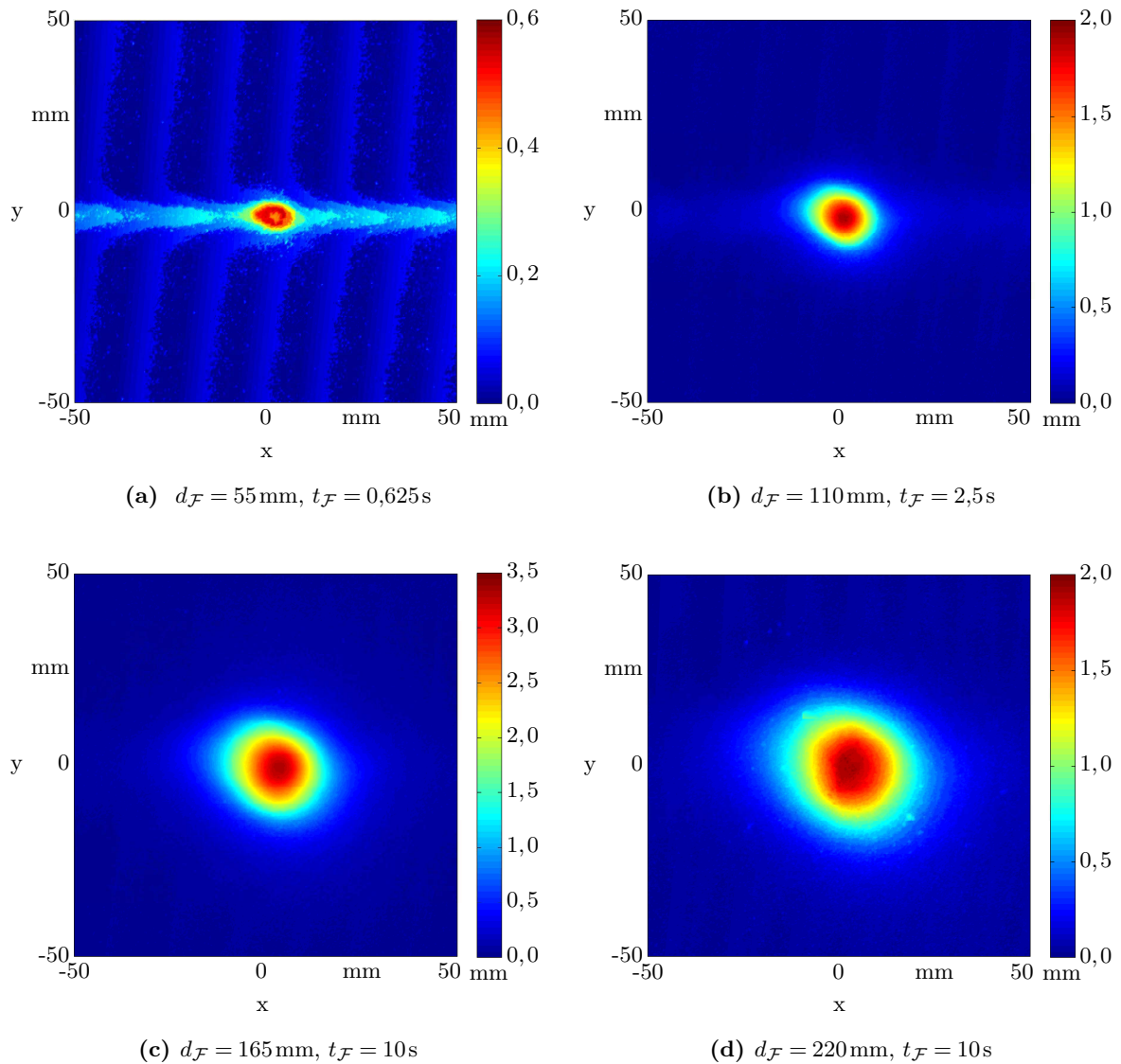


Abbildung 3.39: Ermittelte Footprintprofile der zweiten verwendeten Messreihe (vergleiche Kapitel 3.2) bei variierendem Spritzabstand $d_{\mathcal{F}}$ und Spritzdauer $t_{\mathcal{F}}$. Deutlich erkennbar sind bei Abbildung (a) die durch den Scanprozess eingebrachten Linienartefakte, welche relativ zur in diesem Fall sehr dünnen Beschichtung deutlich hervortreten. Bei einem Spritzabstand von 220 mm (d) sind einige Einzelpartikel und Agglomerationen erkennbar.

funktion potentiell eine für den Lichtbogenspritzprozess typische Bimodalität abbilden könnte, führt die Verwendung einer größeren Anzahl von Gaußfunktionen als Injektoren oder Elektroden im Spritzbrenner, zu einem nicht mehr physikalisch begründeten Modell. Dennoch kann ein solches Modell eine gute und rauschbefreite Approximation für ein gemessenes Profil liefern und somit Vorteile funktionaler und numerischer Repräsentationen vereinen.

Bedingt durch den in Kapitel 3.4 beschriebenen Herstellungsprozess der Footprintproben, bei dem der Spritzbrenner auf einer horizontal verlaufenden Linearbahn mit hoher Geschwindigkeit in der Substratmitte platziert wird, ist das gesuchte Massenflussprofil mit dem Profil einer Linearbahn überlagert. Dieses ist insbesondere bei den Proben mit geringem Spritzabstand und geringer Footprintspritzdauer sichtbar (Abbildungen 3.39a und 3.39b). Daher bietet es sich an, dieses Profil als Summand in das Fittingmodell zu integrieren und nach dem Anpassen des Modells zu entfernen. Das Linearbahnprofil kann dabei gemäß Gleichung (3.9) als Integration über den noch zu bestimmenden Footprint $\mathcal{G}_n(x, y)$ berechnet werden. Es ergibt sich mit $m = 1$ Überlauf, einer normierten Spritzzeit von $t_{\mathcal{F}} = 1$ s sowie angepasstem Integrationsintervall das kombinierte Modell

$$\begin{aligned}
 \mathcal{G}_n^+(x, y) &= \mathcal{G}_n(x, y) + \frac{1}{v} \int_{-\infty}^{\infty} \mathcal{G}_n(x, y) dx \\
 &= \mathcal{G}_n(x, y) + \frac{1}{v} \sum_{i=1}^n \sqrt{2\pi} \sigma_i a_i \cdot e^{-\left(\frac{(y-y_i)^2}{2\tau_i^2}\right)} \\
 &= \sum_{i=1}^n a_i \left(e^{-\left(\frac{(x-x_i)^2}{2\sigma_i^2} + \frac{(y-y_i)^2}{2\tau_i^2}\right)} + \frac{1}{v} \sqrt{2\pi} \sigma_i \cdot e^{-\left(\frac{(y-y_i)^2}{2\tau_i^2}\right)} \right). \quad (3.16)
 \end{aligned}$$

Zusätzlich zu den Amplituden, Mittelpunkten und Standardabweichungen der gesuchten Gaußfunktionen tritt hier die Geschwindigkeit v auf, welche für die Approximation als weiterer unbekannter Koeffizient behandelt oder aber als bekannt angenommen werden kann. So wurde bei der Herstellung der Profile aus Abbildung 3.39 für die Positionierungsgeschwindigkeit v des Roboters 200 mm/s eingestellt und v kann dementsprechend fest definiert werden. Dass diese Vorgabe vom Roboter tatsächlich erreicht wird, kann jedoch nicht garantiert werden. Zumindest am Substratmittelpunkt entstehen in jedem Fall Abweichungen von der programmierten Bahngeschwindigkeit durch das Abbremsen und erneute Beschleunigen des Roboters. Theoretisch könnten diese Abweichungen beispielsweise auf Grundlage einer Dynamiksimulation des Roboters in das Fittingmodell integriert werden. Da jedoch keine verlässlichen Geschwindigkeitsverläufe vorlagen, wurde an dieser Stelle lediglich untersucht, ob die Approximationsgenauigkeit durch eine vom Fittingalgorithmus frei wählbare, entlang der Bahn konstante Geschwindigkeit v gegenüber einer fest vorgegebenen Geschwindigkeit verbessert werden kann. Zur Berechnung der Approximationen wurde ein in der Programmierumgebung Matlab verfügbarer gewichteter nichtlinearer Least-Squares-Ansatz verwendet, welcher Ausreißer automatisch identifiziert und ausschließt.

3.9.2 Eignung der Modellfunktionen

Zur Untersuchung der Eignung der Modellfunktionen wurden verschiedene Approximationen für die vier in Abbildung 3.39 dargestellten Footprintprofile mit unterschiedlichen Spritzabständen, welche zuvor auf eine Spritzdauer von einer Sekunde normiert wurden, berechnet. Dabei wurde zum einen die Anzahl n der überlagerten Gaußfunktionen variiert und zum anderen wurden drei verschiedene Modelle angewandt: \mathcal{G}_n ohne Berücksichtigung der überlagerten Linearbahn sowie \mathcal{G}_n^+ in den zuvor diskutierten Varianten mit v als freiem oder festem Parameter.

Die Abbildungstafel 3.40 zeigt die gerichteten Profilabweichungen zwischen berechneten Gaußmodellen \mathcal{G}_n^+ mit $n = 1, 2, 3, 10, 15$ und 20 überlagerten Einzelfunktionen (Zeilen) für die vier unterschiedlichen Profile (Spalten) mit einer auf 200 mm/s festgelegten Verfahrensgeschwindigkeit. Positive Werte entsprechen dabei einer Unterschätzung des gemessenen Profils durch das Modell, negative Werte einer Überschätzung. Abbildung 3.41 veranschaulicht zusätzlich 90%-Volumenkonturen der berechneten Footprintrepräsentationen als rot eingezeichnete Linien. Als eine $p\%$ -Volumenkontur wird dabei diejenige Kontur konstanter Profilhöhe bezeichnet, welche p Prozent des jeweiligen Gesamtvolumens des Profils einschließt. Bezogen auf den Spritzstrahl kann die 90%-Volumenkontur daher, unter Vernachlässigung der Porosität, als Form des Strahlquerschnitts bei dem jeweiligen Spritzabstand bezeichnet werden. Zusätzlich sind auch die 90%-Volumenkonturen für jede der n Einzelgaußfunktionen eingezeichnet, wobei dunklere Konturen zu Gaußfunktionen mit größerer Amplitude gehören. Die gemäß des zweiten Summanden in Gleichung (3.16) enthaltenen Linearbahnanteile wurden zuvor bereits entfernt und sind nicht Teil der dargestellten Profile.

Bezogen auf die in Abbildung 3.40 dargestellten Abweichungen kann zunächst festgestellt werden, dass die Abweichungen zwischen Scandaten und Fittings bei Spritzabständen von 55 mm und 110 mm größer sind als bei den höheren Abständen. Insbesondere bei $d_{\mathcal{F}} = 55 \text{ mm}$, aber auch noch stellenweise bei 110 mm , sind das im Vergleich zur dünnen Schichtdicke relativ starke Rauschen sowie linienhafte Scanartefakte, welche aus einer leichten Verkipfung der Proben in Wechselwirkung mit der Auflösungsgrenze des Scanners resultieren, sichtbar. Problematisch ist bei dem kürzesten Spritzabstand weiterhin die aufgrund einer Spritzdauer von nur $0,625 \text{ s}$ mangelhafte Separierbarkeit der Linearbahn vom kaum höheren tatsächlichen Footprint. Dies wird in der linken Spalte von Abbildung 3.41 deutlich: Trotz der bereits im Modell \mathcal{G}_n^+ berücksichtigten Linearbahn deckt einer der in die Approximation einfließenden Einzelgaußfunktionen stets die gesamte Linearbahn ab. Bei nur einer Gaußfunktion ($n = 1$) wird somit der eigentliche Footprint überhaupt nicht erfasst, was die hohen Abweichungen in der Mitte des zugehörigen Abweichungsbildes erklärt. Ab $n = 2$ deckt mindestens einer der Summanden auch den eigentlichen Footprintteil ab. Dieses Problem ist für die anderen Proben ab 110 mm Spritzabstand nicht mehr feststellbar, da hier die Spritzdauer mit $2,5\text{--}10 \text{ s}$ deutlich höher ist.

Das Problem der inkorrekten Separation von Footprint und Bahn durch das Fitting tritt auch bei freier Bestimmung der Bahngeschwindigkeit durch den Fittingalgorithmus auf. Die dabei

3. ERZEUGUNG, DIGITALISIERUNG UND NACHBEARBEITUNG VON FOOTPRINTPROFILIEN

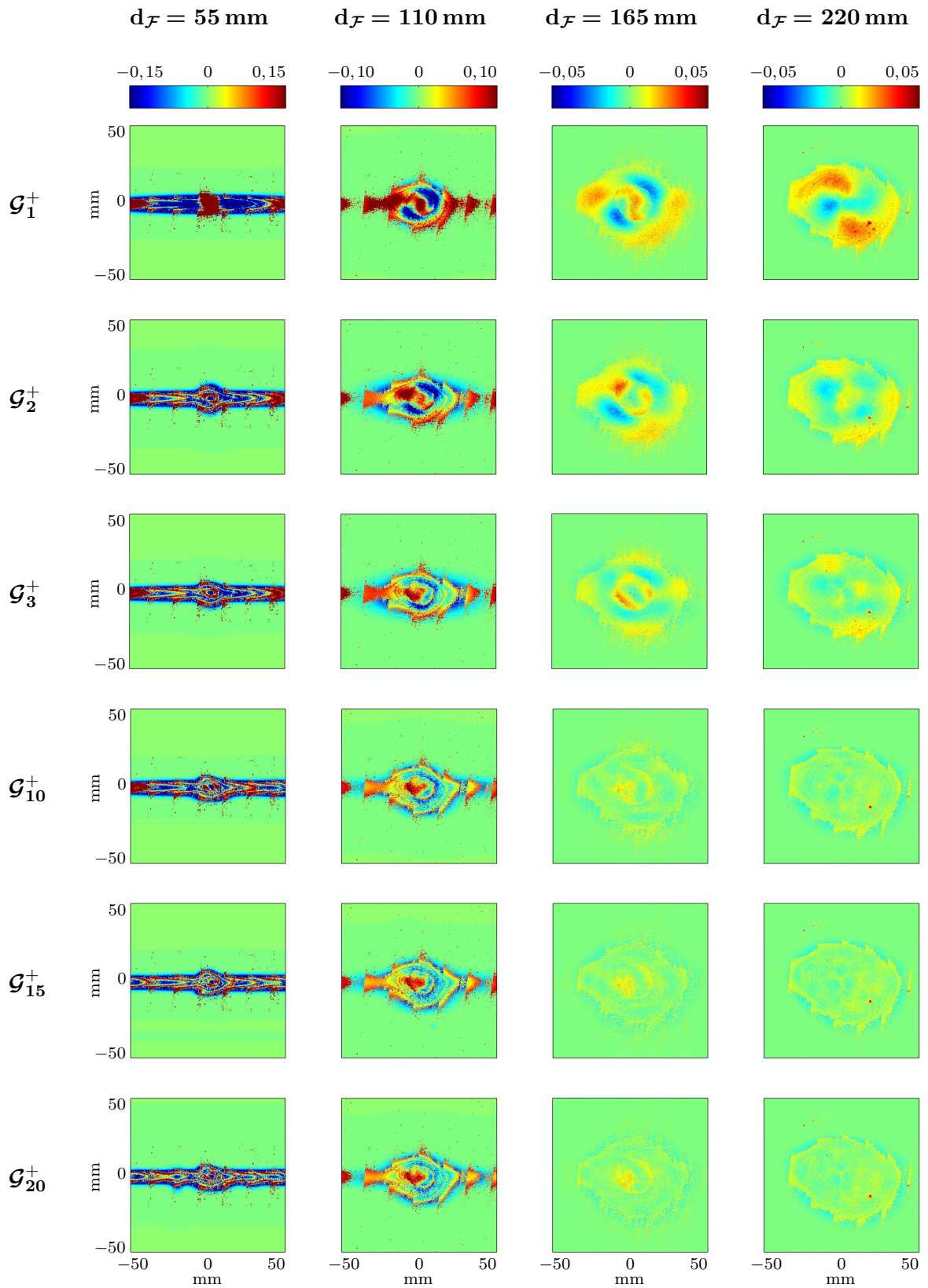


Abbildung 3.40: Abweichung der Modelle \mathcal{G}_n^+ von den Digitalisierdaten (Abweichungen in mm).

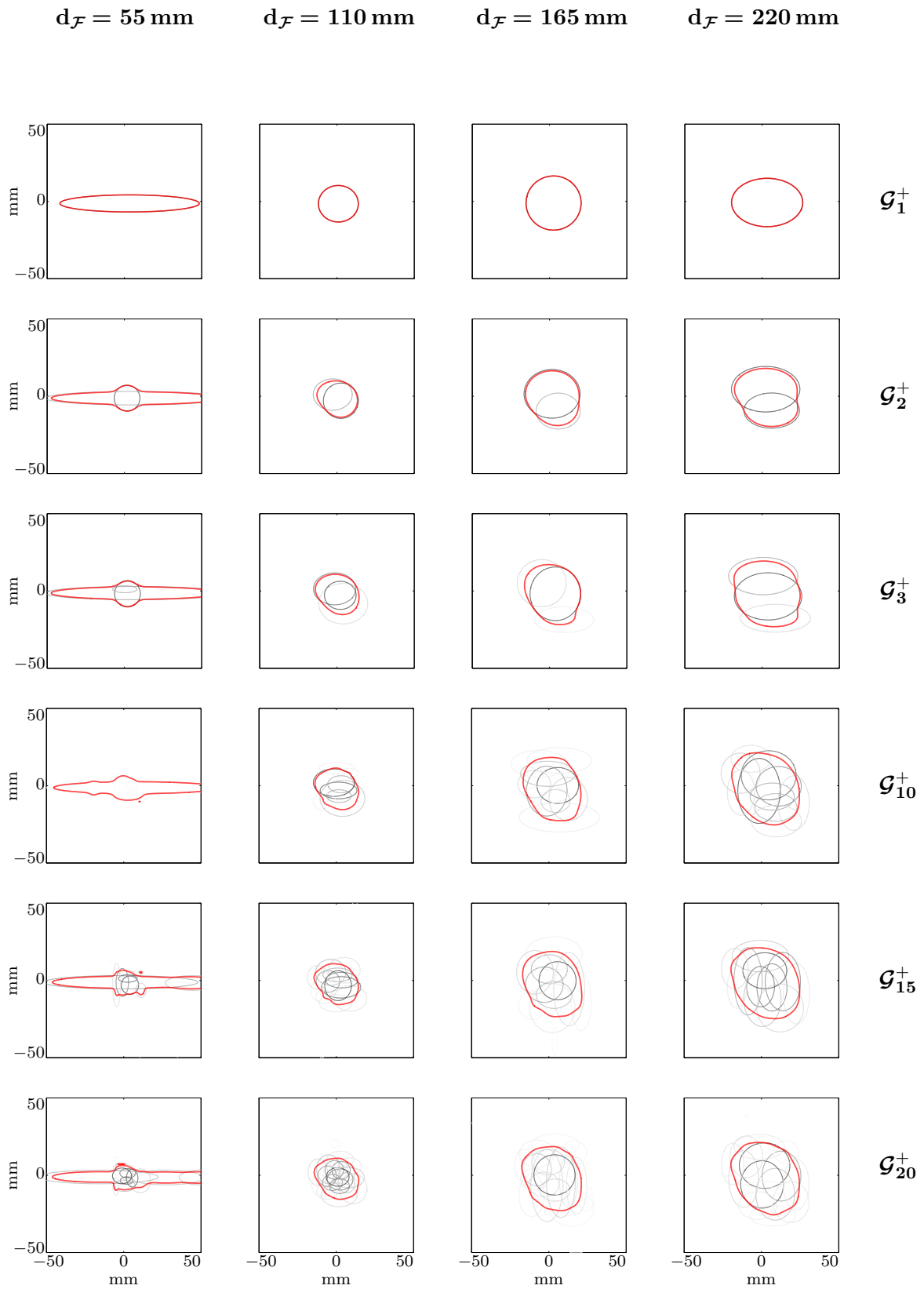


Abbildung 3.41: 90 %-Volumenkonturen der Modelle \mathcal{G}_n^+ sowie der 1–20 Einzelgaußfunktionen.

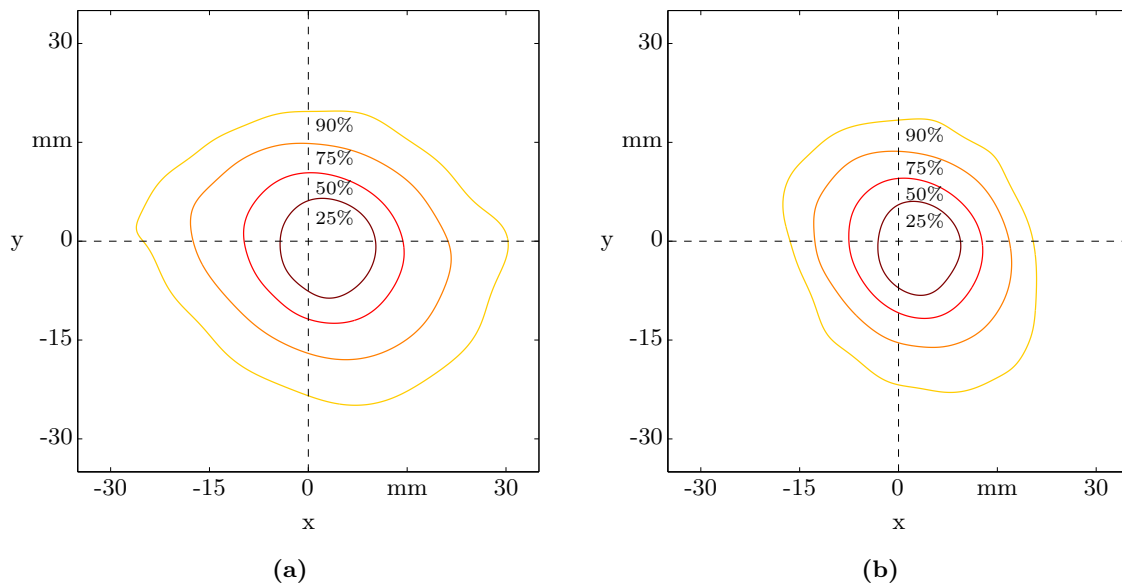


Abbildung 3.42: Konturen der resultierenden Footprints \mathcal{G}_{20}^+ für (a) durch den Fittingalgorithmus bestimmte Verfahrensgeschwindigkeit (≈ 1.667 mm/s) sowie (b) mit festgelegter Verfahrensgeschwindigkeit (200 mm/s). Bei niedrigerer Verfahrensgeschwindigkeit ist der waagrecht verlaufende Linearbahnanteil höher und das verbleibende „tatsächliche“ Footprintprofil schmaler.

für die unterschiedlichen Profile berechneten Geschwindigkeiten weichen meist stark von den tatsächlichen Werten ab und nehmen in der Regel mit steigendem Spritzabstand zu. Dies ist vermutlich darauf zurückzuführen, dass die Linearbahnprofile an den Probenrandbereichen mit steigender Entfernung so dünn werden, dass sie nicht mehr vom Rauschen unterscheidbar sind und daher zu großen Teilen nicht in den Fittingprozess einfließen. Das somit kaum noch vorhandene Linearbahnprofil lässt sich innerhalb des Modells \mathcal{G}_n^+ nur durch sehr hohe Bahngeschwindigkeiten abbilden. Diese wiederum haben jedoch zur Folge, dass die in der Mitte mit dem Footprintanteil überlagerten Linearbahnanteile nicht entfernt, sondern dem Footprint zugerechnet werden. Dementsprechend ergeben sich tendenziell breitere Footprintapproximationen, wie anhand der Volumenkonturen in Abbildung 3.42 zu erkennen ist. Die Verwendung einer festen Bahngeschwindigkeit von 200 mm/s ist daher vorzuziehen. Lediglich für das Profil mit Spritzabstand 55 mm ergibt sich hier mit einer sehr geringen Geschwindigkeit eine vorteilhafte Abtrennung des Footprints von der Linearbahn, welche mit fester Geschwindigkeit nicht möglich war. Diese ist jedoch als nicht fundiert anzusehen und wird daher im Folgenden zwar für die unkritische Visualisierung der Strahlgeometrie, aber nicht für deren Analyse oder die Simulation verwendet.

Das Modell \mathcal{G}_n ohne Berücksichtigung der Linearbahnen aus Gleichung (3.16) führt zu Ergebnissen, welche in Bezug auf die Separation von Bahn und Footprint beziehungsweise auf die Breite des resultierenden Footprints der Variante mit nicht festgesetzter Geschwindigkeit stark ähneln. Gegenüber dem Modell mit fester Geschwindigkeit sind die Gesamtvolumina der

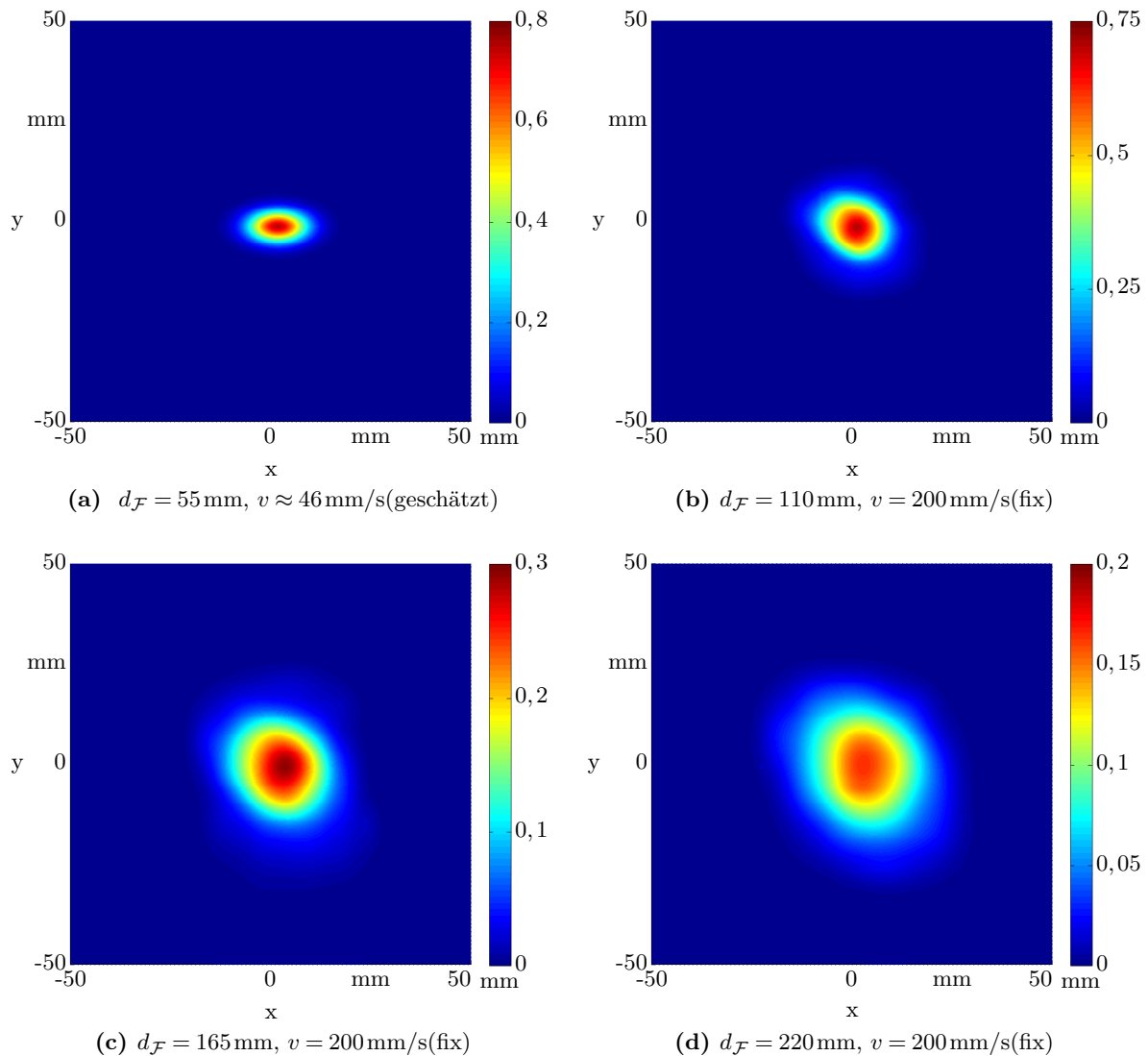


Abbildung 3.43: Ermittelte Footprintrepräsentationen für die Spritzabstände 55–220 mm basierend auf den Modellen (a) \mathcal{G}_1^+ mit geschätzter bzw. (b)–(d) \mathcal{G}_{20}^+ mit fixer Verfahrensgeschwindigkeit. Aufgrund der Subtraktion der überlagerten Linearbahnanteile weisen die Profile im Vergleich zu den Profilen in Abbildung 3.39 eine etwas schmalere Charakteristik auf.

Footprints aufgrund der mit in den Footprint einbezogenen Linearbahnanteile erwartungsgemäß durchgehend größer. Abgesehen von dem etwas geringeren Berechnungsaufwand birgt das Modell keine weiteren Vorteile.

In Bezug auf die zu verwendende Anzahl n der Summanden im Modell kann festgestellt werden, dass ab $n = 10$ kaum noch sichtbare und nur marginal messbare Verbesserungen der Abweichungen erreicht werden. Gegenüber den physikalisch motivierbaren ein oder zwei Gaußfunktionen erzielen zehn oder mehr Summanden jedoch sichtbar geringere Abweichungen. Aus den sich ähnelnden Fehlerverteilungen für Modelle mit ein bis drei Gaußfunktionen ist ersichtlich, dass zumindest einfache funktionale Approximationen, wie sie häufig verwen-

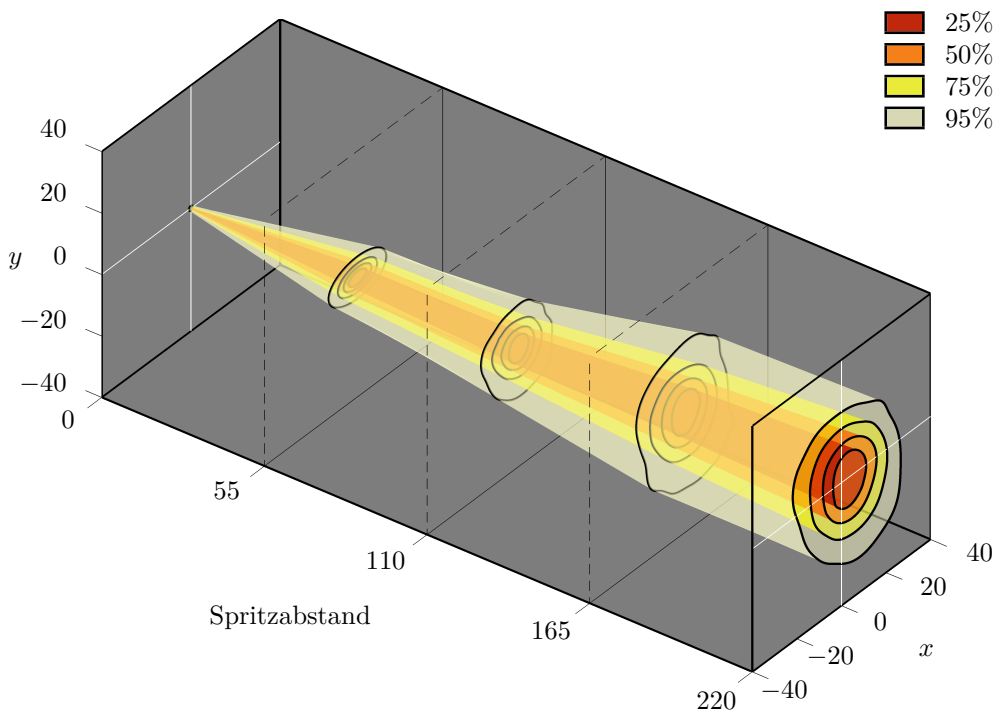


Abbildung 3.44: Visualisierung der Spritzstrahlgeometrie basierend auf Volumenflusskonturen, welche 25–95 % des Volumenflusses einschließen. Als Grundlage zur Berechnung dienen die funktionsbasierten Footprintrepräsentationen \mathcal{G}_n^+ bei Spritzabständen von 55–220 mm. Für die Düse wurde eine kreisrunde Öffnung mit einem Durchmesser von ca. 1 cm angenommen.

det werden, zu keinem guten Ergebnis führen. Im Folgenden werden daher die berechneten Footprintmodelle \mathcal{G}_{20}^+ für $d_{\mathcal{F}} = 110\text{--}220\text{ mm}$ verwendet, da diese die genaueste Approximation liefern, auch wenn die Unterschiede zu \mathcal{G}_{10}^+ marginal sind (Abbildung 3.43). Diese Profile werden entsprechend ihres Spritzabstandes mit \mathcal{F}_{110} , \mathcal{F}_{165} und \mathcal{F}_{220} bezeichnet. Ausschließlich zu Visualisierungszwecken wird ferner das mit \mathcal{F}_{55} bezeichnete Profil im Spritzabstand 55 mm verwendet, welches aus dem Fitting \mathcal{G}_1^+ mit variabler Bahngeschwindigkeit hervorging.

3.9.3 Ergebnisse zur funktionsbasierten Profilrepräsentation

Auf Grundlage der ermittelten Footprintprofile in unterschiedlichen Spritzabständen lässt sich nun die Spritzstrahlgeometrie analysieren. Von besonderem Interesse für die Simulation ist dabei die Frage, welchen Öffnungswinkel der Spritzstrahl aufweist und ob die Simulationsannahme einer linearen Zunahme des Footprintdurchmessers mit dem Spritzabstand, also eines entfernungsunabhängigen Öffnungswinkels, gerechtfertigt ist.

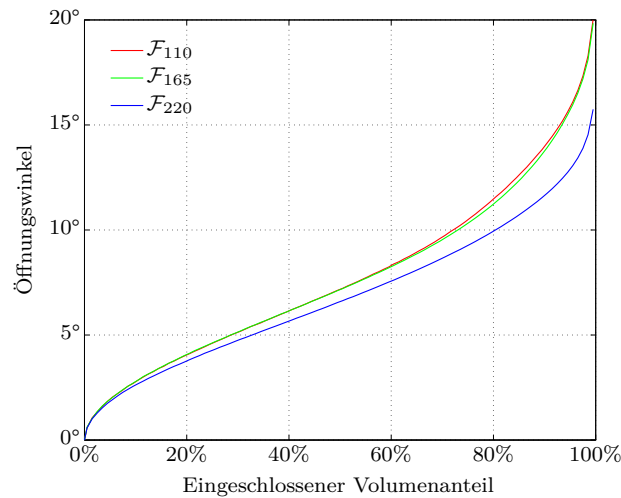


Abbildung 3.46: Verlauf der Strahlöffnungswinkel in Abhängigkeit des eingeschlossenen Footprintvolumens für die drei Spritzabstände 110 mm, 165 mm und 220 mm.

bis zur 50 %-Grenze bei einem Spritzabstand von 220 mm markieren, kann – mit Ausnahme der als kritisch anzusehenden Footprintmessung bei 55 mm – eine gute Übereinstimmung mit der Kontur im Bereich zwischen \mathcal{F}_{110} und \mathcal{F}_{220} festgestellt werden. Wie bereits Djurić und Grant festgestellt haben, ist eine eindeutige Bestimmung eines Öffnungswinkels zur Charakterisierung des Spritzstrahls schwierig [45], da keine scharf abgezeichnete Strahlgrenze existiert. Um die für die Beschichtungssimulation wichtige Annahme linearer Partikelflugbahnen beziehungsweise eines vom Spritzabstand unabhängigen Öffnungswinkels zu untermauern, werden daher an dieser Stelle die gesamten Verläufe der durchschnittlichen Öffnungswinkel in Abhängigkeit des Anteils des eingeschlossenen Footprintvolumens für die verschiedenen Spritzabstände betrachtet. Diese sind in Abbildung 3.46 dargestellt und wurden für jede funktionale Footprintrepräsentation berechnet, indem zunächst die Volumenkontur konstanter Schichtdicke ermittelt wurde, welche den entsprechenden Volumenanteil umfasst. Für alle Punkte auf dieser Kontur wurde anschließend der Öffnungswinkel in Relation zum Düsenmittelpunkt berechnet und der Mittelwert aller Öffnungswinkel entlang der Kontur gebildet.

In der Abbildung wird deutlich, dass der Verlauf der Öffnungswinkel für verschiedene Volumenanteile für \mathcal{F}_{110} und \mathcal{F}_{165} fast exakt übereinstimmt. Für \mathcal{F}_{220} ergibt sich ein ähnlicher Verlauf, die Öffnungswinkel sind jedoch insbesondere im Randbereich des Footprints im Vergleich etwas kleiner. Diese Abweichungen könnten hier auf eine leichte Einschnürung des Spritzstrahls, aber auch auf Messfehler in den Randbereichen des Footprints, welche eine Schichtdicke im Bereich der Oberflächenrauheit durch das Sandstrahlen aufweisen, hindeuten. Aufgrund des in guter Näherung linearen Verlaufs des Strahldurchmessers zwischen der Düse und den Profilen \mathcal{F}_{110} und \mathcal{F}_{165} erscheint die Annahme eines kegelförmigen Spritzstrahls mit linear im Abstand ansteigendem Durchmesser zumindest in diesem für den Prozess relevanten Bereich gerechtfertigt. Ob es sich bei einem Abstand von 220 mm um eine leichte Einschnürung und keine Messabweichung handelt, ließe sich nur durch weitere Versuche klären.

Zusätzlich zur Betrachtung des Öffnungswinkels liefern die mit zunehmendem Abstand vom Spritzbrenner abnehmenden Volumina der ermittelten Profile auch einen Anhaltspunkt zur Abhängigkeit der Auftragsrate oder Porosität vom Spritzabstand. So sinkt das Volumen der Footprints von 148 mm^3 bei 110 mm Abstand zunächst auf $140,1 \text{ mm}^3$ für \mathcal{F}_{165} und dann auf $122,4 \text{ mm}^3$ für \mathcal{F}_{220} ab. Diese Beobachtung deckt sich mit der geringfügigen Abnahme des im Experiment gemessenen Gewichts der Profile. Auch von Krebs [85] wurde bei der Untersuchung desselben Schichtsystems eine mit zunehmendem Spritzabstand geringfügig abnehmende Porosität und Auftragseffizienz festgestellt, welche jedoch beide großen Schwankungen unterliegen. Ausgehend von einer Auftragsrate von 100% bei einem Abstand von $d = 110 \text{ mm}$ ergibt sich anhand der Footprintprofile als bestmögliche lineare Approximation die prozentuale Auftragseffizienz

$$E(d) = 118,4 - 0.1577 \cdot d \%. \quad (3.17)$$

Vereinigung der Einzelprofile

Für die Verwendung der ermittelten Profildaten im Rahmen einer Beschichtungssimulation kann der Schichtauftrag an einem Oberflächenpunkt, welcher sich innerhalb des Spritzkegels befindet, prinzipiell durch Interpolation zwischen den beiden nächstliegenden Footprintrepräsentationen vor und hinter dem Oberflächenpunkt oder anhand eines einzigen auf den Punkt projizierten Profilvertes berechnet werden. Bei Verwendung eines einzelnen Profils, wie im Fall der in Kapitel 5 vorgestellten Simulation, ist es sinnvoll, die vorliegenden Informationen für alle Spritzabstände zu einem Durchschnittsprofil zusammenzufassen. Dafür werden die Einzelprofile zunächst auf eine einzige Distanz von der Düse umgerechnet, indem die Achsen der Footprints entsprechend des Spritzabstandes skaliert und die Profildicken gemäß der geänderten Grundfläche des Footprints angepasst werden. Für die Umrechnung auf die Entfernung 110 mm ergibt sich der Durchschnittsfootprint zu

$$\bar{\mathcal{F}}_{110} = \left(\mathcal{F}_{110}(x, y) + m_1 s_1^2 \cdot \mathcal{F}_{165}(s_1 x, s_1 y) + m_2 s_2^2 \cdot \mathcal{F}_{220}(s_2 x, s_2 y) \right) / 3.$$

Darin sind $s_1 = 165/110$ und $s_2 = 220/110$ Skalierungsfaktoren zur Umrechnung des Spritzabstandes und m_2 und m_3 Faktoren zur Vereinheitlichung des Profilvervolumens auf das Volumen von \mathcal{F}_{110} . Letztere sind erforderlich, um in der Simulation mittels Gleichung (3.17) die Auftragseffizienz ausgehend vom Abstand 110 mm zu berücksichtigen. Abbildung 3.47 zeigt den resultierenden Durchschnittsfootprint $\bar{\mathcal{F}}_{110}$ sowie die relativen Abweichungen der Footprints \mathcal{F}_{110} , \mathcal{F}_{165} und \mathcal{F}_{220} von diesem Durchschnittsfootprint. Diese wurden basierend auf $e_{\text{rel}}(x, y)$ aus Gleichung (3.5) mittels

$$e_{\text{relV}}(p) = \frac{1}{|\Theta_p|} \sum_{(x,y) \in \Theta_p} e_{\text{rel}}(x, y) \quad (3.18)$$

berechnet. Die Menge Θ_p beinhaltet dabei die Messpunkte innerhalb der $p\%$ -Volumenkontur und beschränkt so den Bereich, in dem die durchschnittliche Abweichung bestimmt wird. Aufgrund des nach außen stark abnehmenden Volumenflusses steigt der relative Fehler erwartungsgemäß ab ca. $p = 90\%$ stark an. Unterhalb dieser Grenze liegen die Abweichungen

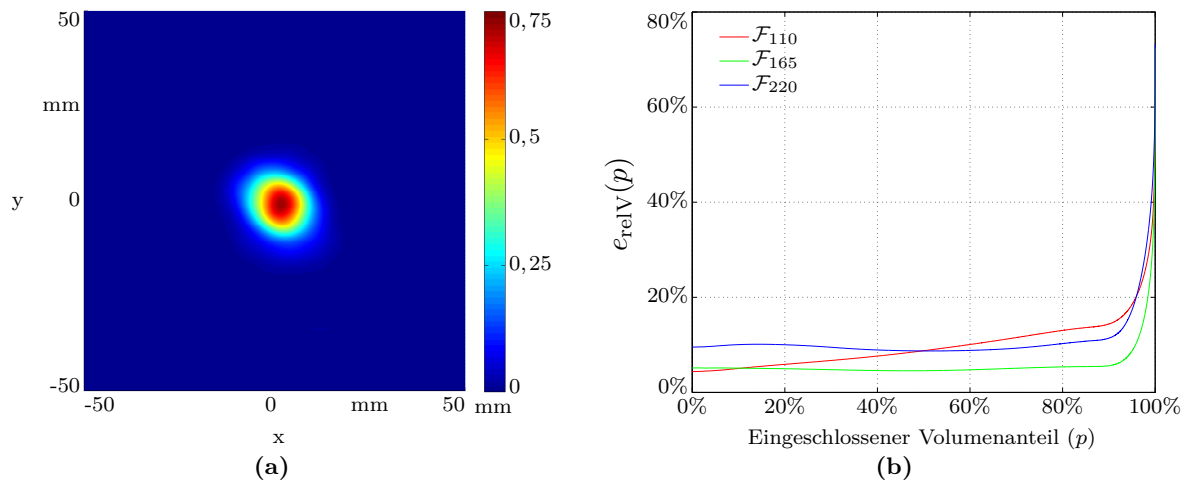


Abbildung 3.47: (a) Durchschnittlicher Footprint $\bar{\mathcal{F}}_{110}$. (b) Abweichungen der Footprintmodelle verschiedener Spritzabstände von diesem Footprint $\bar{\mathcal{F}}_{110}$ in Abhängigkeit des eingeschlossenen Footprintvolumens nach Gleichung (3.18).

zwischen 5 % und 10 %. Diese können als Indikator für die Stabilität des Prozesses und zur Einschätzung der zu erwartenden maximalen Genauigkeit der Simulation dienen.

Beschichtungsmodell

Ein Beschichtungsmodell ist eine mathematische Beschreibung des Masse- bzw. Volumentransfers vom Spritzbrenner auf die Oberfläche eines Werkstücks. Vereinfacht ausgedrückt gibt das Beschichtungsmodell vor, wie viel Spritzmaterial pro Zeiteinheit in welche Richtung vom Brenner abgegeben wird, welcher Anteil davon zu jedem Zeitpunkt der Simulation jeden Oberflächenpunkt des Werkstücks erreicht, wie viel des Materials auf der Werkstückoberfläche haften bleibt und welche Schichtdickenzunahme sich aus der resultierenden Materialmenge ergibt. In der Simulation wird durch das Beschichtungsmodell die Charakteristik des zu simulierenden Prozesses definiert. Die durch ein festgelegtes Beschichtungsmodell insgesamt abgebildete Charakteristik ist dabei in der Regel spezifisch für einen einzigen Prozess und eine einzige Parametrisierung dieses Prozesses. Das bedeutet, dass sowohl ein Wechsel beispielsweise von einem Lichtbogen- zu einem HVOF-Spritzprozess als auch eine Änderung wichtiger Prozessparameter, wie beispielsweise dem Primär- oder Sekundärgasdruck, eine Anpassung des Modells erfordern. Um verschiedene Prozesse und Parametrisierungen simulativ abbilden zu können, ohne ein grundlegend neues Modell zu entwerfen und zu implementieren, ist es daher sinnvoll, die Modellierung in verschiedene Teilkomponenten aufzuteilen. Dadurch können prozessunabhängige von prozessabhängigen Aspekten separiert und im Rahmen des Simulationsframeworks allgemeingültig und effizienter implementiert werden. Das Beschichtungsmodell für das im Rahmen dieser Arbeit entwickelte Simulationssystem setzt sich aus folgenden Komponenten zusammen:

1. grundlegende Massen- oder Volumenstromdichteverteilung
2. geometrische Übertragungsfunktion
3. Haftwirkungsgrad und winkelabhängige Porosität.

Die Massenstromdichteverteilung definiert die von einem (virtuellen) Spritzbrenner richtungsabhängig pro Zeiteinheit emittierte Materialmenge und ist somit vom Prozess und seiner Parametrisierung abhängig. Darauf aufbauend spezifiziert die geometrische Übertragungsfunktion die Berechnungsvorschrift zur Ermittlung des Materialanteils, der jeden Punkt der Werkstückoberfläche erreicht, sowie zur Bestimmung des daraus resultierenden Schichtdickenzuwachses. Die aufgestellte Übertragungsfunktion bildet ausschließlich von der Art des Spritzprozesses oder seiner Parametrisierung unabhängige, geometrische Zusammenhänge ab und kann daher allgemeingültig implementiert werden. Die dritte Komponente modelliert den Haftwirkungsgrad (engl.: sticking efficiency) und gibt an, welcher Anteil der auf die Werkstückoberfläche auftreffenden Partikel dort haften bleibt und nicht wieder abprallt. Der Haftwirkungsgrad ist von vielen Faktoren wie Materialeigenschaften des Spritzwerkstoffs, kinetischer und thermischer Energie der Partikel, Oberflächeneigenschaften des Substrates und dem Einschlagwinkel abhängig.

Die detaillierte Modellierung dieser drei verschiedenen Komponenten wird in den folgenden Abschnitten 4.2, 4.3 und 4.4 beschrieben. Im folgenden Abschnitt 4.1 werden jedoch zunächst in der Literatur vorhandene Modelle betrachtet.

4.1 Beschichtungsmodelle in der Literatur

Zur Simulation thermischer Spritzprozesse oder des verwandten und bereits in Kapitel 2.3.2 erläuterten Sprühkompaktierprozesses existieren in der wissenschaftlichen Literatur eine Reihe von Veröffentlichungen, welche Beschichtungsmodelle explizit oder implizit zur Schichtdickenberechnung nutzen. In der überwiegenden Mehrheit der Arbeiten kommt dabei ein als rotationssymmetrisch und gaußförmig angenommenes Spritzprofil oder geringfügige Abwandlungen hiervon zum Einsatz [15, 18, 33, 34, 42, 69, 89, 92, 98, 121, 132, 157–159, 177].

Ausführlich beschrieben wird diese Modellierung unter anderem von Seok et al. [132], welche zunächst eine Volumenstromdichte, gemessen in $\text{mm}^3/(\text{mm}^2 \text{s})$, der Form

$$\dot{M}_{\text{seok}}(r) = a \cdot \exp(-b \cdot r^2)$$

annehmen. Dabei stellt r die kürzeste Distanz eines Raumpunktes zur Spritz- und Symmetrieachse des Brenners dar. Die Parameter a und b hängen vom Gesamtvolumenstrom, gemessen in mm^3/s , so wie vom Spitzabstand d ab, welcher der Entfernung vom nächstgelegenen Punkt auf der Spritzachse zur Düse entspricht. Der Gesamtvolumenstrom kann für diesen Ansatz durch Integration der Volumenstromdichte in einer zur Spritzrichtung orthogonalen Ebene in Polarkoordinaten (r, θ) ermittelt werden:

$$\bar{M}_{\text{seok}} = \int_0^\infty \int_0^{2\pi} a \cdot \exp(-b \cdot r^2) \cdot r d\theta dr = \frac{\pi a}{b}.$$

Das Verhältnis der Parameter a und b ist also für einen konstanten Volumenstrom für alle Distanzen d zur Düse fix. Unter der Annahme des Volumenerhalts im divergierenden Spritz-

strahl ergibt sich für zwei Distanzen d_1 und d_2 die Beziehung [132]

$$\frac{a_1}{a_2} = \frac{b_1}{b_2} = \left(\frac{d_2}{d_1}\right)^2 \quad (4.1)$$

für die Parameter der Volumenstromdichte a_1 und b_1 beziehungsweise a_2 und b_2 beider Distanzen. Anschaulich bedeutet dies, dass der Maximalbetrag a der gaußförmigen Volumenstromdichte mit zunehmendem Abstand von der Düse quadratisch abnimmt, da sich der Volumenstrom auf eine quadratisch anwachsende Fläche verteilt. Ebenso steigt mit zunehmendem Abstand die zu b invers proportionale Streuung σ quadratisch an, was einer Verbreiterung des Volumenstroms entspricht.

Wird nun der Volumenstrom durch experimentelle Messungen von abgelagerten Spritzprofilen in einem Abstand d_1 gemessen, können die Parameter a_1 und a_2 durch ein Fitting der Modellfunktion auf die gemessenen Daten ermittelt werden. In der Simulation können dann für einen Raumpunkt mit Spritzabstand d_2 zunächst mittels Gleichung 4.1 die Parameter a_2 und b_2 und anschließend die lokale Volumenstromdichte $\dot{M}_{\text{seok}}(r)$ ermittelt werden. Zumeist erfolgt dieser Berechnungsschritt implizit und wird nicht wie in Kapitel 4.3 dieser Dissertation als separater Schritt zur expliziten Entkoppelung der grundlegenden Volumenstrommodellierung von den geometrischen Bezügen in der Simulation aufgeführt. Zur Berechnung des lokalen Schichtwachstums in einem Zeitintervall dt wird $\dot{M}_{\text{seok}}(r)$ auf die lokale Oberflächennormale projiziert und mit dem Zeitschritt sowie separat modellierten Funktionen für den Haftwirkungsgrad gewichtet.

Diesem Modell folgen die Publikationen [33, 34, 69, 92, 98, 177]. Die Arbeiten [15, 18, 42, 157–159] verwenden ebenfalls gaußförmige Profile ohne explizit auf diese Modellierung einzugehen. Bis zu drei Gaußfunktionen, welche drei Pulverinjektoren des Spritzbrenners modellieren, werden in [121] berücksichtigt. Leigh und Berndt [89] erweitern diese analytische Formulierung um einen Anstellwinkel des Spritzbrenners.

Djurić et al. [44–47] verwenden abweichend von der üblichen Formulierung keine gaußförmigen, sondern mittels der Kosinusfunktion modellierte Volumenstromdichten der Arten

$$\dot{M}_{\text{djurić}}(\theta) = A(\gamma_g) \cdot \cos\left(\frac{\pi}{2\gamma_g}\theta\right), \quad \theta \in [0, \gamma_g]$$

und

$$\dot{M}_{\text{djurić}_2}(\theta) = \frac{1}{2}A_2(\gamma_g) \left(1 + \cos\left(\frac{\pi}{\gamma_g}\theta\right)\right), \quad \theta \in [0, \gamma_g],$$

wobei θ der Winkel zwischen der Spritzachse und der Richtung von der Düse zu einem betrachteten Punkt und γ_g ein zu definierender Öffnungshalbwinkel des Spritzkegels ist. Die Faktoren A und A_2 werden volumenerhaltend definiert [44]. Eine zu $\dot{M}_{\text{djurić}}(\theta)$ vergleichbare Formulierung nutzt auch [48].

Fasching et al. [54] sowie Lugscheider et al. [95, 96] nutzen keine analytische Formulierung, sondern akkumulieren direkt gemessene zweidimensionale beziehungsweise mittels einer Freistrahlsimulation berechnete Profile.

Bis auf die drei letztgenannten Arbeiten, welche lediglich eine zweidimensionale Modellierung vornehmen oder auf einer externen Freistrahlsimulation basieren, verwenden alle genannten Publikationen rotationssymmetrische Spritzcharakteristiken. Durch das in diesem Kapitel vorgestellte dreidimensionale und Footprint-basierte Modell sollen diese Beschränkungen umgangen und die Abbildung beliebiger Spritzcharakteristiken auf Basis gemessener Footprintprofile realisiert werden.

4.2 Massenstromdichteverteilung

Die Modellierung des von einem Spritzbrenner richtungsabhängig emittierten Massenflusses ist von zentraler Bedeutung für die korrekte Abbildung eines Prozesses in der Simulation. Diese Massenflussverteilung ist nicht nur von der Art des Prozesses (zum Beispiel Lichtbogenspritzen, Hochgeschwindigkeits-Flammspritzen oder Plasmaspritzen) abhängig, sondern auch von der Parametrisierung des verwendeten Spritzprozesses. So bedingen beispielsweise veränderte Gasdrücke beim Lichtbogenspritzprozess eine Auffächerung oder Fokussierung des Spritzstrahls; Änderungen an Elektrodenspannung oder -strom verändern die Drahtabschmelzgeschwindigkeit und somit den Gesamtmassenstrom, also die insgesamt pro Sekunde abgegebene Masse.

Da der Massentransport in unterschiedlichen Spritzprozessen von verschiedenen Parametern auf komplexe Weise beeinflusst wird, ist es nicht sinnvoll, die Wirkung einzelner Parameter verschiedener Prozesse in einem generischen Simulationsmodell abzubilden. Stattdessen dient eine einheitliche, für unterschiedliche Prozesse anwendbare Modellierung des Massentransportes in Form einer Massenstromdichteverteilung als Grundlage für die Simulation. Diese Verteilungsfunktion kann als minimale gemeinsame Modellierungsschnittstelle zwischen verschiedenen Spritzprozessen auf der einen und der Simulation der Schichtablagerung auf der anderen Seite angesehen werden. Diese Modellierung des Massenflusses geht mittels einschränkender Annahmen aus einer allgemeinen Betrachtung der volumetrischen Strömung hervor, welche auf dem Gebiet des Freistrahls definiert ist.

4.2.1 Physikalische Modellierung als volumetrische Strömung

Aus physikalischer bzw. fluiddynamischer Sicht kann der vom Spritzbrenner ausgehende Massenstrom im allgemeinen Fall durch die Betrachtung der (Massen-)Dichteverteilung

$$\rho(\mathbf{x}, t) \tag{4.2}$$

sowie des Geschwindigkeitsvektorfeldes

$$\mathbf{v}(\mathbf{x}, t) = (v_x(\mathbf{x}, t), v_y(\mathbf{x}, t), v_z(\mathbf{x}, t))^T, \tag{4.3}$$

welches die Strömungsgeschwindigkeit und Richtung in jedem Punkt \mathbf{x} zu jedem Zeitpunkt t angibt, charakterisiert werden. Da der partikelbeladene Spritzstrahl in der Regel ein mehrphasiges Gemisch aus gasförmigen Brenn- oder Zerstäubergasen sowie flüssigen und festen

Materialpartikeln darstellt, ist für eine umfassende Darstellung die separate Betrachtung der Dichteverteilungen und Geschwindigkeitsfelder für die einzelnen Phasen üblich [12, 84]. Eine getrennte Betrachtung der Phasen ist insbesondere für die Simulation des Ablöse- oder Beschleunigungsverhaltens einzelner Partikel, welche von der schnelleren Gasphase mitgerissen werden, sowie der Berechnung ihrer Trajektorien von Interesse, da hier Kräfte an den Grenzflächen zwischen unterschiedlichen Phasen von Bedeutung sind [153]. Für die Beschichtungssimulation genügt jedoch die Betrachtung des Dichte- und Geschwindigkeitsfeldes des Spritzmaterials, da die Gasphase keinen direkten materiellen Beitrag zur Schichtbildung liefert.

Zur Herleitung der auf einer Oberfläche in einer bestimmten Zeit abgelagerten Masse wird zunächst der Materialtransport durch eine virtuelle, in der Strömung platzierte Fläche A betrachtet. Im Fall einer planaren, an jeder Stelle senkrecht und mit konstanter Geschwindigkeit $v = |\mathbf{v}|$ durchströmten Fläche ergibt sich das in der Zeit dt durch A geschobene Volumen $A \cdot v \cdot dt$ der Höhe $v \cdot dt$ über A . Im Fall einer konstanten Dichteverteilung ist die Gesamtmasse dieses Volumens $\rho \cdot v \cdot A \cdot dt$ und die zeitliche Normierung liefert den Massenstrom $\rho \cdot A \cdot v$, also die pro Zeiteinheit durch A strömende Masse. Für eine beliebig geformte und in beliebiger Weise durchströmte Fläche ergibt sich der Massenstrom durch die Fläche als Oberflächenintegral zweiter Art:

$$\dot{m}_A = \iint_A \rho \mathbf{v} \cdot d\mathbf{A} = \iint_A \mathbf{j} \cdot d\mathbf{A} = \iint_A \langle \mathbf{j}, \mathbf{n}_A \rangle dA. \quad (4.4)$$

Darin ist \mathbf{n}_A der Einheitsnormalenvektor der Fläche A und $\mathbf{j} = \rho \mathbf{v}$ wird als Massenstromdichte bezeichnet, welche den Massenstrom pro Fläche repräsentiert und daher in der Einheit $\text{kg}/(\text{m}^2 \text{s})$ angegeben werden kann. Durch die skalare Multiplikation der Massenstromdichte mit dem Flächennormalenvektor werden hier nur Massenstromanteile senkrecht zu A erfasst, parallele Anteile tragen nicht zum Gesamtmassenstrom bei. Die in einem Zeitintervall $[t_1, t_2]$ durch A strömende Masse ergibt sich schließlich durch Zeitintegration zu

$$m_A = \int_{t_1}^{t_2} \iint_A \mathbf{j} \cdot d\mathbf{A} dt. \quad (4.5)$$

In Analogie zu dieser Berechnung der Masse durch zeitliche Integration des Massenstroms, welcher wiederum durch Integration über einer Querschnittsfläche aus der Massenstromdichte hervorgeht, können auch Volumen, Volumenströme und Volumenstromdichten definiert werden. Dabei wird auf die Multiplikation mit der Dichte ρ verzichtet. Der Volumenstrom, gemessen in m^3/s , ergibt sich dann direkt aus der Integration der Geschwindigkeit über A :

$$\dot{V}_A = \iint_A \mathbf{v} \cdot d\mathbf{A}. \quad (4.6)$$

Das in einer bestimmten Zeit durch A strömende Volumen ergibt sich analog zu Gleichung (4.5) durch Zeitintegration zu

$$V_A = \int_{t_1}^{t_2} \iint_A \mathbf{v} \cdot d\mathbf{A} dt \quad (4.7)$$

und das Vektorfeld der „Volumenstromdichte“ ist nichts anderes als das Geschwindigkeitsfeld $\mathbf{v}(\mathbf{x}, t)$ selbst.

Diese Betrachtung als Volumenstrom erleichtert das Verständnis des Zusammenhangs dieser Strömungsformulierung mit dem Schichtwachstum. Betrachtet man ein infinitesimales, orthogonal zur Strömungsrichtung platziertes Oberflächenelement dA und das in der infinitesimalen Zeitspanne dt darauf abgelagerte Volumen V_{dA} , so ist die Höhe des abgelagerten Volumens V_{dA}/dA . Nach Gleichung (4.7) wächst die Schicht in diesem Fall mit einer Momentangeschwindigkeit von

$$\frac{V_{dA}}{dA dt} = |\mathbf{v}|,$$

da $dA \perp \mathbf{v}$. Im allgemeinen Fall einer nicht orthogonalen Durchströmung von dA kann $-\mathbf{v}(\mathbf{x}, t)$ als momentane und nicht unbedingt normale Bewegungsgeschwindigkeit eines Oberflächenpunktes \mathbf{x} aufgefasst werden [45]. Im Beschichtungsprozess bewegen sich die Oberflächenpunkte der wachsenden Schicht also zu jedem Zeitpunkt entgegen der Stromlinien des Geschwindigkeitsfeldes in Richtung des Spritzbrenners. Dabei ist zu beachten, dass die Wachstumsgeschwindigkeit $|\mathbf{v}|$ der Schicht nicht mit der Strömungsgeschwindigkeit des mehrphasigen Stoffgemischs im Spritzstrahl identisch ist, sondern lediglich die Geschwindigkeit einer äquivalenten einphasigen Strömung des reinen Spritzmaterials ohne gasförmige Anteile angibt.

Da die Formulierungen für die massen- oder volumenbasierte Strömung bis auf den konstanten Faktor ρ identisch sind, macht es aus algorithmischer Sicht nahezu keinen Unterschied, welche Betrachtungsweise verwendet wird. Relevant wird dieser erst mit der Modellierung der in der abgelagerten Schicht enthaltenen Porosität, welche typischerweise primär in Abhängigkeit des Auftreffwinkels $\angle(\mathbf{v}, -\mathbf{n}_A)$ berechnet wird. Wird mittels einer Massenstrom-basierten Formulierung die an einer Stelle des Bauteils abgelagerte Masse m_A berechnet, muss bei einer Ableitung der Schichtdicke aus der Masse m_A die enthaltene Porosität berücksichtigt werden. Aus der Masse m_A kann also nicht auf ein lokal abgelagertes Volumen von m_a/ρ geschlossen werden. Der Vorteil einer Volumenstrom-basierten Sichtweise besteht an dieser Stelle darin, dass formal davon ausgegangen wird, dass sich das abgelagerte Volumen direkt aus dem mathematisch beschriebenen Volumenstromfeld ergibt. Für die Schichtdickenberechnung ist somit keine gesonderte Porositätsberechnung erforderlich, wenn diese anhand gemessener Volumenprofile von Footprints und anderen Versuchen kalibriert wird. Für die im Rahmen dieser Arbeit entwickelte Simulation ist daher eine Modellierung der mit dem Auftreffwinkel variierenden Porosität implizit in der Modellierung des ebenfalls winkelabhängigen Haftwirkungsgrades enthalten, welcher basierend auf dem abgelagerten Schichtvolumen verschiedener Proben bestimmt wird (vergleiche Kapitel 4.4.1).

Ein praktischer Vorteil der Modellierung als Massenstrom kann allerdings darin bestehen, dass die Massenförderrate \dot{M} insbesondere bei pulverbasierten Spritzverfahren leicht gemessen oder direkt maschinell vorgegeben werden kann. Aufgrund der Massenerhaltung gilt nach dem Gaußschen Integralsatz für jede beliebige geschlossene Hüllfläche A^* , die den Brenner umschließt [102],

$$\dot{M} = \oint_{A^*} \mathbf{j} \cdot d\mathbf{A}. \quad (4.8)$$

Diese Eigenschaft erlaubt für eine Massenstrom-basierte Modellierung die Skalierung aufgestellter Massenstromdichteverteilungen anhand einfach ermittelbarer Messdaten oder Maschinenparameter für \dot{M} . Für das im Rahmen dieser Arbeit primär verwandte Lichtbogenspritzverfahren ist die Förderrate jedoch nicht direkt einstellbar, sondern ergibt sich indirekt aus der eingestellten Leistung des Spritzbrenners in Form von Spannung und Stromstärke. Die Förderrate der Spritzdrähte wird dabei durch einen maschineninternen Regelkreis bestimmt, welcher diese zu jedem Zeitpunkt genau so schnell zuführt, dass sich ein stabiles Abschmelzverhalten einstellt. Ohne externe Messtechnik kann auch die Vorschubrate nicht bestimmt werden, so dass eine Massenstrom-basierte Modellierung hier keinen Vorteil bietet.

4.2.2 Footprintexperimente

Die Formulierung des Massentransfers ist in der vorgestellten allgemeinen Form nicht ohne weitere Einschränkungen in einer Simulation nutzbar. Ein wesentliches Problem besteht in diesem Kontext darin, dass die grundlegenden zeitabhängigen Dichte- und Geschwindigkeitsfelder, ρ und \mathbf{v} , nicht bekannt sind und auch messtechnisch weder vollständig noch mit einer für die Simulation ausreichenden Genauigkeit erfasst werden können. Lediglich durch eine Simulation des Freistrahls könnten vollständige Felder berechnet werden. Dies steht allerdings der praktischen Einsetzbarkeit der entwickelten Simulation für unterschiedliche Prozesse und Parametrisierungen entgegen, da für jeden Einzelfall eine Freistrahlsimulation entwickelt werden müsste, um den veränderlichen Parametern Rechnung zu tragen. Darüber hinaus liefern die aus der Literatur bekannten Ansätze, wie in Kapitel 3.1 dargestellt ist, noch keine zufriedenstellende Genauigkeit.

Experimentell kann der Massenstrom mit ausreichender Genauigkeit zwar nicht volumetrisch, aber in einer Querschnittsfläche zur Spitzrichtung gemessen werden, indem der Spritzstrahl an einer definierten Stelle durch ein in den Strahl eingebrachtes, planares Substrat unterbrochen und die auf dem Substrat abgelagerte Schicht vermessen wird. Das vermessene Profil wird als Footprintprofil bezeichnet und ist genau das den Prozess charakterisierende Footprintprofil, dessen Erzeugung, Digitalisierung und Nachbearbeitung in Kapitel 3 ausführlich behandelt wurde. Aufgrund seiner zentralen Bedeutung für das Beschichtungsmodell wird im Folgenden die Definition aus Kapitel 3 noch einmal wiederholt:

Definition. *Unter einem **Footprint** \mathcal{F} wird dasjenige dreidimensionale Profil des beim thermischen Spritzen abgelagerten Spritzmaterials verstanden, welches entsteht, indem unter definiertem (Raum-)Winkel, Abstand und für eine definierte Zeit auf ein initial ebenes, relativ zur Spritzpistole unbeweglich positioniertes Substrat gespritzt wird.*

Formal dargestellt wird ein Footprintprofil $\mathcal{F}(x, y)$ im Folgenden als Höhenprofil über der x, y -Ebene des Substrats. Der auf das Intervall $(-1, 1)^2 \subset \mathbb{R}^2$ normierte Definitionsbereich von \mathcal{F} wird als Footprintkoordinatensystem bezeichnet (vergleiche Kapitel 6), dessen Ursprung im Schnittpunkt der Substratebene mit der die Spritzrichtung definierenden z -Achse des Werkzeugkoordinatensystems liegt.

Der zwischen Oberflächennormalen und Spritzrichtung gemessene Spritzwinkel beträgt dabei 0° und der Abstand entspricht in der Regel dem Standardarbeitsabstand des Prozesses, also 110 mm für den verwendeten Lichtbogenspritzprozess. Die Spritzdauer beträgt für den Lichtbogenspritzprozess bis zu 10 s und für andere Prozesse bis zu 45 s. Sowohl der Spritzabstand als auch die Dauer werden im Rahmen der entwickelten Beschichtungssimulation als Zusatzinformationen zu jedem Footprintprofil mitgeführt, um korrekte Schichtdicken berechnen zu können.

4.2.3 Modellannahmen

Um die Volumen- bzw. Massenstromdichteverteilung und damit den Spritzprozess mittels eines experimentell ermittelten Footprintprofils beschreiben zu können, ist es erforderlich, dass dieses Profil repräsentativ für den Prozess ist. In der Praxis ist daher dafür Sorge zu tragen, dass der Prozess eine möglichst gute Stabilität und Wiederholgenauigkeit aufweist und dass bei der Anfertigung der Footprintprofile dieselben Bedingungen wie im Produktionseinsatz vorliegen. Zusätzlich sollte die Spritzdauer $t_{\mathcal{F}}$ für die Anfertigung der Footprintprofile hinreichend groß gewählt werden, so dass kurzzeitige Prozessschwankungen, etwa durch ungleichmäßige Draht- oder Pulverförderung, nicht ins Gewicht fallen. Für die Simulation wird daher davon ausgegangen, dass das Footprintprofil das gemittelte Prozessverhalten wiedergibt und sich der Prozess durch einen als stationär angenommenen Massenstrom beschreiben lässt, also Dichte- und Geschwindigkeitsfelder nicht zeitabhängig sind.

Um den Volumen- oder Massenstrom an jeder Stelle im Raum berechnen zu können, wird des Weiteren davon ausgegangen, dass der Massenstrom von einer Punktquelle¹ an der Düse des Brenners ausgeht und sich geradlinig ausbreitet. Für die Simulation thermischer Spritzprozesse ist dies eine übliche Annahme, welche darin begründet ist, dass die Trägheitskräfte einzelner Partikel im Spritzstrahl aufgrund der in der Regel hohen Dichte des Spritzmaterials und der hohen Partikelgeschwindigkeiten dominieren und die Partikeltrajektorien als näherungsweise linear angenommen werden können [7, 15, 27, 44, 69, 103, 132].

Nicht unbedingt gültig ist die Annahme geradliniger Trajektorien in der initialen Beschleunigungsphase der Partikel [6]. Für pulverbasierte Verfahren ist dies die Zone in der die Partikel, deren initiale Flugrichtung vom Pulverinjektor bestimmt wird, vom Gasstrom mitgerissen und in Spritzrichtung umgelenkt werden. Für den im Rahmen dieser Arbeit primär betrachteten Lichtbogenspritzprozess ermittelten Tillmann et al. das Vorhandensein von Turbulenzen bis zu einer Entfernung von 50 mm vor der Düse aufgrund von Verdichtungsstößen in diesem Bereich [153]. Die Partikelgeschwindigkeiten fielen in diesem Bereich zunächst leicht ab, erreichten aber bereits im Bereich bis 80 mm, welcher unterhalb des für den Spritzprozess relevanten Entfernungsbereiches liegt, ihre Maximalgeschwindigkeit. Auch die im Rahmen dieser Arbeit untersuchte und in Kapitel 3.9.3 dokumentierte Unabhängigkeit des Strahlöffnungswinkels von der Spritzdistanz deutet darauf hin, dass für die relevanten Spritzabstände im Bereich

¹Für eine zylindrische Spritzstrahlcharakteristik, welche in Kapitel 4.3.4 diskutiert wird, liegt die Punktquelle im Grenzübergang unendlich weit hinter dem Brenner und der Massenstrom ist nicht divergent.

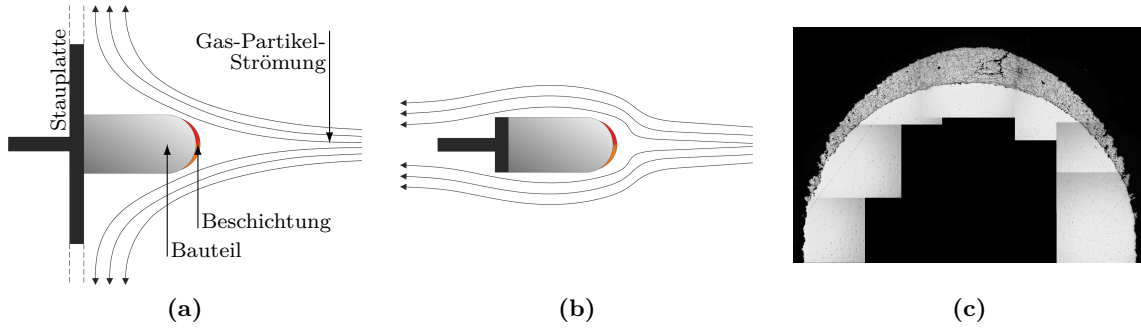


Abbildung 4.1: Zur Untersuchung von Turbulenzen vor dem Substrat wurden verschiedene Außenradialproben beschichtet, wobei die Gasströmung optional durch eine Stauplatte abgelenkt wurde. (a) und (b) Aufbau mit und ohne Stauplatte. (c) Lichtmikroskopischer Querschliff einer Probe mit einem Radius von 5 mm. Durch eine Stauung der Strömung konnten keine signifikanten Unterschiede in der resultierenden Beschichtung festgestellt werden.

um 110 mm von linearen Partikeltrajektorien ausgegangen werden kann.

Des Weiteren treten Turbulenzen in der Gasströmung, welche die Partikelflugbahnen beeinflussen könnten, im Bereich unmittelbar vor dem Substrat auf, da hier die Gasströmung unterbrochen wird. Ba et al. haben den Effekt verschieden gekrümmter Substrate auf die Flugbahnen von Partikeln verschiedener Größe im Plasmaspritzprozess numerisch untersucht und festgestellt, dass lediglich Partikel mit einem Durchmesser von deutlich unter $10\ \mu\text{m}$ von Strömungsänderungen vor dem Substrat beeinflussbar sind [5]. Für das Lichtbogenspritzverfahren ist die Partikelgrößenverteilung jedoch das Resultat eines komplexen Ablösungsprozesses von den Elektroden und dem nachfolgenden Zerfallen der Partikel im turbulenten Zerstäuber gas und daher nicht auf einfache Weise vorhersagbar. Ob die Partikelablagerung bei diesem Verfahren quantitativ durch eine Stauung des Spritzstrahls beeinflussbar ist, wurde anhand einer Versuchsreihe, bei der halbzyklindrische Außenradialen mit Durchmessern im Bereich von 5 bis 40 mm beschichtet wurden, untersucht. Die Stauung des Strahls wurde dabei durch eine ausreichend große, direkt hinter der Probe angebrachte Stahlplatte bewirkt, wie in Abbildung 4.1 dargestellt ist. Ein Vergleich der erzielten Schichtdicken mit und ohne Stauplatte anhand lichtmikroskopischer Querschliffe zeigte keine Unterschiede in der Schichtablagerung, welche über zu erwartende Prozessschwankungen und Messungenauigkeiten hinausgeht.

Die Annahme, dass das Geschwindigkeitsfeld \mathbf{v} radial und geradlinig von der Quelle ausgeht, ist also zumindest für den untersuchten und für den Prozess relevanten Bereich gerechtfertigt. In Polarkoordinaten (r, ϑ, φ) mit der Punktquelle im Ursprung vereinfacht sich damit die Darstellung des stationären Geschwindigkeitsfeldes für $r > 0$ zu

$$\mathbf{v}(\mathbf{r}, t) = \mathbf{v}(\mathbf{r}) = \mathbf{v}(r, \vartheta, \varphi) = \tilde{v}(r, \vartheta, \varphi) \frac{\mathbf{r}}{|\mathbf{r}|},$$

wobei $\mathbf{r} = (r \sin \vartheta \cos \varphi, r \sin \vartheta \sin \varphi, r \cos \vartheta)^T$. Um der Massenerhaltung nach Gleichung (4.8) zu genügen, muss der Betrag der Massenstromdichte ferner entlang $\mathbf{r}/|\mathbf{r}|$ quadratisch mit dem Abstand abnehmen, da die durchströmte Strahlquerschnittsfläche quadratisch ansteigt.

Es gilt daher

$$\mathbf{v}(\mathbf{r}) = \frac{1}{r^2} v(\vartheta, \varphi) \frac{\mathbf{r}}{|\mathbf{r}|}. \quad (4.9)$$

4.3 Geometrische Übertragungsfunktion

Die im vorherigen Kapitel getroffenen Modellannahmen ermöglichen nun die Beschreibung der Materialübertragung auf das Bauteil durch Angabe eines Footprintprofils $\mathcal{F}(x, y)$ sowie der zugehörigen Footprintspritzdauer $t_{\mathcal{F}}$ und Spritzentfernung $d_{\mathcal{F}}$ von der Düse des Brenners zum ebenen Footprintsubstrat. Auf Basis dieser Angaben kann mittels einer prozessunabhängigen, rein geometrisch definierten Übertragungsfunktion das in einem Simulationszeitschritt Δt an einem Punkt auf der Bauteiloberfläche resultierende Schichtwachstum berechnet werden. Die Herleitung dieser Übertragungsfunktion wird im Folgenden beschrieben, wobei zunächst nur der Volumenstrom unter Vernachlässigung des Haftwirkungsgrades betrachtet wird. Formulierungen für den Massenstrom ergeben sich vollständig analog.

4.3.1 Kegelförmiger Spritzstrahl

Abbildung 4.2a zeigt eine vereinfachte zweidimensionale Darstellung der geometrischen Zusammenhänge für die im Folgenden betrachtete dreidimensionale Beschichtungssituation. Dabei sind sowohl die Werkstückoberfläche, für die der Schichtzuwachs bestimmt werden soll, als auch die Position des Footprintsubstrates entsprechend des durchgeführten Footprintexperiments eingezeichnet, um die geometrischen Zusammenhänge beider Vorgänge zu verdeutlichen. Die Betrachtung erfolgt im kartesischen Spritzkegelkoordinatensystem, welches seinen Ursprung in der Spitze des in orange angedeuteten Spritzkegels hat und so ausgerichtet ist, dass die negative z -Koordinatenrichtung der Spritzrichtung entspricht und die x - und y -Koordinatenachsen parallel zum Footprintsubstrat verlaufen. Die Wahl dieses Koordinatensystems basiert auf dem OpenGL-Kamerakoordinatensystem, welches für die in Kapitel 5 beschriebene Implementierung zum Einsatz kommt.

Betrachtet wird nun ein Punkt mit Ortsvektor $\mathbf{a} = (a_x, a_y, a_z)^T$ auf dem Footprintsubstrat und ein Punkt $\alpha \mathbf{a}$ ($\alpha > 0$) auf der Werkstückoberfläche, welcher auf dem gleichen vom Ursprung ausgehenden Strahl liegt. Der Volumenstrom entlang dieses Strahls bestimmt den Schichtdickenzuwachs sowohl auf dem Footprintsubstrat am Punkt \mathbf{a} als auch auf dem Werkstück am Punkt $\alpha \mathbf{a}$. Fasst man den Strahl als infinitesimal schmales kegel- oder pyramidenförmiges Volumen auf, welches die Footprintebene im Bereich eines in \mathbf{a} zentrierten, infinitesimalen Flächenelements $dA_{\mathcal{F}}$ und die lokal als planar angenommene Werkstückoberfläche im Flächenelement $dA_{\mathcal{W}}$ in der Umgebung von $\alpha \mathbf{a}$ durchdringt, so ist der Volumenstrom durch $dA_{\mathcal{F}}$ und $dA_{\mathcal{W}}$ identisch (Abbildung 4.2). Dies folgt unmittelbar aus dem Gaußschen Integralsatz nach Gleichung (4.8) angewandt auf den Volumenfluss durch die Hüllfläche des Volumenelements sowie aus der zuvor getroffenen Annahme, dass sich die Strömung wie in Gleichung (4.9) beschrieben radial geradlinig ausbreitet. Die Volumenstromdichte \mathbf{v} verläuft

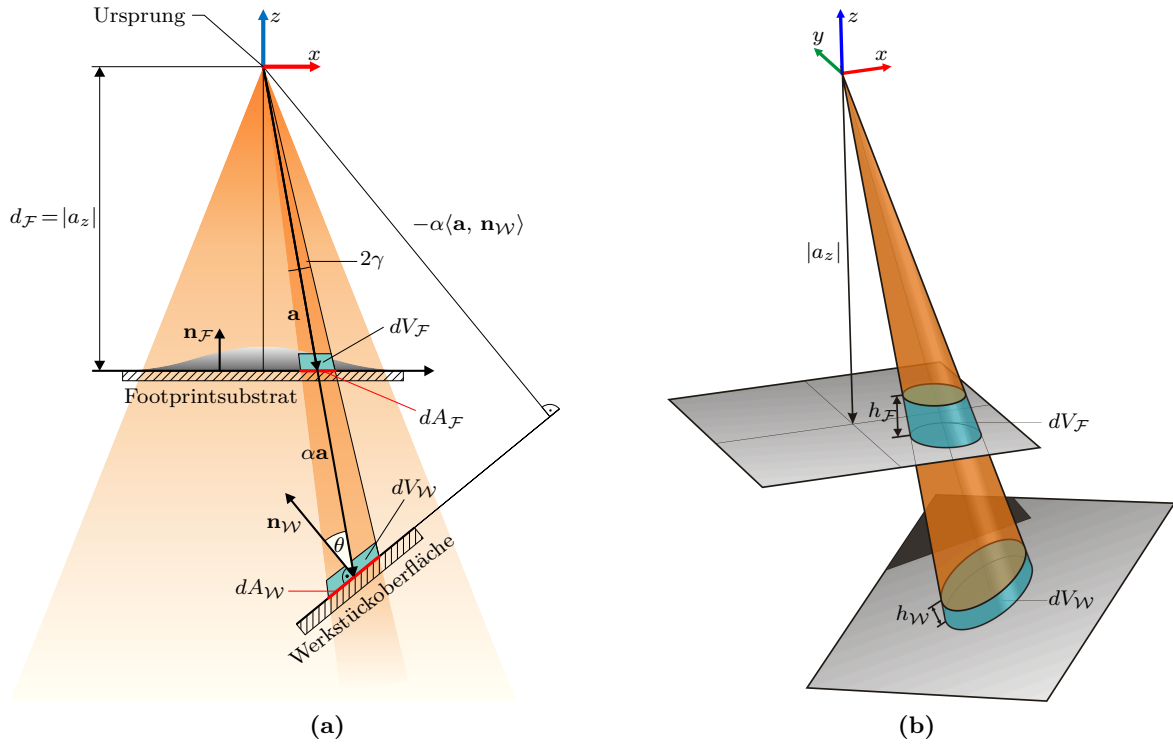


Abbildung 4.2: Geometrische Zusammenhänge zwischen Spritzstrahl, Footprintexperiment und Beschichtungssimulation für eine beliebige Werkstückoberfläche. (a) Zweidimensionaler Querschnitt durch den Spritzstrahl. (b) Dreidimensionale Darstellung des betrachteten infinitesimalen Strahlvolumens. Ziel ist die Berechnung der Höhe des Pyramidenstumpfes $h_{\mathcal{W}}$ auf der Werkstückoberfläche auf Basis der Footprintprofilhöhe $h_{\mathcal{F}}$.

daher stets parallel zur Mantelfläche A_M des Volumenelements, so dass durch A_M keine Strömung stattfindet:

$$\dot{V}_M = \oint_{A_M} \mathbf{v} \cdot d\mathbf{A}_M = 0.$$

Der gesamte Volumenstrom durch jede beliebige lückenlose Querschnittsfläche des volumetrischen Strahls, wie $dA_{\mathcal{F}}$ oder $dA_{\mathcal{W}}$, ist daher

$$\dot{V} = \oint_{dA_{\mathcal{F}}} \mathbf{v} \cdot d\mathbf{A}_{\mathcal{F}} = \oint_{dA_{\mathcal{W}}} \mathbf{v} \cdot d\mathbf{A}_{\mathcal{W}}.$$

Die Tatsache, dass damit auch das pro Zeiteinheit durch $dA_{\mathcal{W}}$ strömende Volumen $dV_{\mathcal{W}}$ identisch zu dem in derselben Zeitspanne durch $dA_{\mathcal{F}}$ strömenden Volumen $dV_{\mathcal{F}}$ ist, wird im Folgenden genutzt, um eine Formulierung für die auf $dA_{\mathcal{W}}$ abgelagerte Schichtdicke auf Basis der bekannten Schichtdicke auf dem Footprintsubstrat zu ermitteln. Im Grenzübergang $dA_{\mathcal{W}} \rightarrow 0$ ergibt sich aus dieser Formulierung dann ein Ausdruck für das Schichtwachstum an einem Punkt auf der Werkstückoberfläche, welcher in der Simulation implementiert werden kann.

Die am Footprintpunkt \mathbf{a} gemessene Schichtdicke sei $h_{\mathcal{F}}$. Betrachtet werde der Volumenstrom in dem schmalen Kreiskegel mit der Mittelachse $\mathbf{0} + \lambda \mathbf{a}$ ($\lambda \in \mathbb{R}_0^+$) und dem Öffnungswinkel 2γ , wie in Abbildung 4.2 dargestellt. Das in der Zeit $t_{\mathcal{F}}$ durch das elliptische Footprintflächenelement $dA_{\mathcal{F}}$ strömende Volumen kann damit als Volumen eines Kegelstumpfes der Höhe $h_{\mathcal{F}}$ berechnet werden. Mit der Formel $V_{\text{Kegel}} = h \cdot A/3$ für das Volumen eines elliptischen Kegels mit Grundfläche A und Höhe h über A ergibt sich dieses Volumen als Differenz der Volumina zweier Kegel mit Grundflächen $dA_{\mathcal{F}}$ und der um $h_{\mathcal{F}}$ verschobenen Grundfläche $dA_{\mathcal{F}}^*$:

$$dV_{\mathcal{F}} = \frac{1}{3}d_{\mathcal{F}} \cdot dA_{\mathcal{F}} - \frac{1}{3}(d_{\mathcal{F}} - h_{\mathcal{F}}) \cdot dA_{\mathcal{F}}^*. \quad (4.10)$$

Die Fläche $dA_{\mathcal{F}}$ berechnet sich darin als Schnittfläche des Kegels mit der Footprintebene, welche die Kegellachse im Abstand $|\mathbf{a}| = d_{\mathcal{F}}/\cos(\theta_{\mathcal{F}})$ und unter dem Winkel $\theta_{\mathcal{F}}$ schneidet:

$$dA_{\mathcal{F}} = \frac{\pi \cdot d_{\mathcal{F}}^2 \cdot \sin^2(\gamma)}{\cos(\theta_{\mathcal{F}}) \cos(\gamma - \theta_{\mathcal{F}}) \cos(\gamma + \theta_{\mathcal{F}}) \sqrt{1 - \tan^2(\gamma) \tan^2(\theta_{\mathcal{F}})}}. \quad (4.11)$$

Die Herleitung dieser Schnittfläche ist in Anhang A beschrieben. Da die obere Fläche $dA_{\mathcal{F}}^*$ des Kegelstumpfes den Kegel unter gleichem Winkel, jedoch im Abstand $(d_{\mathcal{F}} - h_{\mathcal{F}})/\cos(\theta_{\mathcal{F}})$ von der Kegelspitze schneidet, folgt aus Gleichung (4.11) bzw. (A.3) in Anhang A

$$dA_{\mathcal{F}}^* = \left(\frac{d_{\mathcal{F}} - h_{\mathcal{F}}}{d_{\mathcal{F}}} \right)^2 dA_{\mathcal{F}}. \quad (4.12)$$

Durch Einsetzen von $dA_{\mathcal{F}}^*$ sowie $d_{\mathcal{F}} = -a_z$ in Gleichung (4.10) ergibt sich der Ausdruck

$$dV_{\mathcal{F}} = \frac{1}{3}h_{\mathcal{F}} \left(\left(\frac{a_z + h_{\mathcal{F}}}{a_z} \right)^2 + \left(\frac{a_z + h_{\mathcal{F}}}{a_z} \right) + 1 \right) dA_{\mathcal{F}} \quad (4.13)$$

für das in der Footprintspritzdauer $t_{\mathcal{F}}$ auf der Teilfläche $dA_{\mathcal{F}}$ des Footprintsubstrats abgelagerte Materialvolumen. Dieses Volumen durchdringt in dieser Zeitspanne auch das werkstückseitige Flächenelement $dA_{\mathcal{W}}$ und die gesuchte Höhe dieser Ablagerung ergibt sich wiederum durch die Betrachtung des kegelstumpfförmigen Volumenelements $dV_{\mathcal{W}}$ aus Abbildung 4.2b. Analog zu Gleichung (4.13) ergibt sich mit dem Abstand $-\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle$ der Werkstückoberfläche von der Kegelspitze sowie der ebenfalls analog zu Gleichung (4.11) ermittelten Fläche

$$dA_{\mathcal{W}} = \frac{\pi \cdot (\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle)^2 \cdot \sin^2(\gamma)}{\cos(\theta_{\mathcal{W}}) \cos(\gamma - \theta_{\mathcal{W}}) \cos(\gamma + \theta_{\mathcal{W}}) \sqrt{1 - \tan^2(\gamma) \tan^2(\theta_{\mathcal{W}})}} \quad (4.14)$$

für dieses Volumen die Formulierung

$$dV_{\mathcal{W}} = \frac{1}{3}h_{\mathcal{W}} \left(\left(\frac{\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle + h_{\mathcal{W}}}{\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle} \right)^2 + \left(\frac{\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle + h_{\mathcal{W}}}{\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle} \right) + 1 \right) dA_{\mathcal{W}}. \quad (4.15)$$

Das Auflösen dieser Gleichung nach der gesuchten Höhe liefert zunächst

$$h_{\mathcal{W}} = \sqrt[3]{\alpha^3 \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle^3 + 3\alpha^2 \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle^2 \frac{dV_{\mathcal{W}}}{dA_{\mathcal{W}}} - \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle}. \quad (4.16)$$

Da im Rahmen der Simulation das in einem Zeitintervall Δt erzielte Schichtdickeninkrement bestimmt werden soll, wird in diese Gleichung, aufgrund der angenommenen Stationarität des Prozesses, das aus dem Footprint-seitigen Volumen abgeleitete Ablagerungsvolumen

$$dV_{\mathcal{W}} = \frac{\Delta t}{t_{\mathcal{F}}} dV_{\mathcal{F}} \quad (4.17)$$

eingesetzt. Darin bezeichnet $t_{\mathcal{F}}$ die Spritzdauer bei der Fertigung des Footprints. Einsetzen der Gleichungen (4.11), (4.13), (4.14) und (4.17) in Gleichung (4.16) liefert den umfangreichen Ausdruck

$$h_{\mathcal{W}} = \left[\frac{\langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle^2 h_{\mathcal{F}} (3a_z^2 + 3a_z h_{\mathcal{F}} + h_{\mathcal{F}}^2)}{2a_z^2} \frac{\Delta t}{t_{\mathcal{F}}} \frac{\cos(\theta_{\mathcal{F}}) (\cos(2\gamma) + \cos(2\theta_{\mathcal{W}}))}{\cos(\theta_{\mathcal{W}}) \cos(\gamma - \theta_{\mathcal{F}}) \cos(\gamma + \theta_{\mathcal{F}})} \dots \right. \\ \left. \dots \frac{\sqrt{1 - \tan^2(\gamma) \tan^2(\theta_{\mathcal{W}})}}{\sqrt{1 - \tan^2(\gamma) \tan^2(\theta_{\mathcal{F}})}} + (\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle)^3 \right]^{1/3} - \alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle.$$

Im Grenzübergang $\gamma \rightarrow 0$ zu einem infinitesimal schmalen Kegelelement vereinfacht sich dieser Ausdruck für die in der Zeit Δt an einem Punkt $\alpha \mathbf{a}$ abgelagerte Schichtdicke

$$\lim_{\gamma \rightarrow 0} h_{\mathcal{W}} = \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle \left(\sqrt[3]{\left(\left(1 + \frac{h_{\mathcal{F}}}{a_z} \right)^3 - 1 \right) k_t + \alpha^3 - \alpha} \right), \quad (4.18)$$

wobei $k_t = \Delta t / t_{\mathcal{F}}$ zu einem Zeitfaktor zusammengefasst wurde. Alle auf der rechten Seite auftretenden Parameter sind dabei in der Beschichtungssimulation auf einfache Weise zu ermitteln.

4.3.2 Veranschaulichung der kegelbasierten Übertragungsfunktion

Der Verlauf der geometrischen Übertragungsfunktion (4.18) ist für verschiedene Parameterbelegungen nicht ohne Weiteres aus der Gleichung ersichtlich. Daher wird der Verlauf der Funktion in Abhängigkeit von verschiedenen Parametern, insbesondere dem Abstand von der Werkstückoberfläche und dem Einschlagwinkel $\theta_{\mathcal{W}}$, in diesem Kapitel exemplarisch dargestellt. Darüber hinaus wird im folgenden Kapitel durch eine weitergehende Vereinfachung der Übertragungsfunktion eine Relation zu bekannten Zusammenhängen aus dem Gebiet der Strahlungsübertragung bzw. der Beleuchtungsberechnung hergestellt.

Vor einer Darstellung der Übertragungsfunktion muss zunächst ihr Definitionsbereich betrachtet werden, da nicht alle Parameterkombinationen physikalisch korrekte Schichtdickenwerte liefern. Ungültige Fälle treten auf, wenn das in einem Zeitintervall Δt abgelagerte Volumen größer als das Volumen des gesamten kegelförmigen Volumenausschnitts zwischen der Werkstückoberfläche $dA_{\mathcal{W}}$ und dem Spritzbrenner ist. Dieser Fall tritt auf, wenn der Brenner so nah an der Werkstückoberfläche positioniert ist oder der Simulationszeitschritt Δt so groß gewählt wurde, dass die Beschichtung in einem Zeitschritt bis zum Brenner wächst. Nach

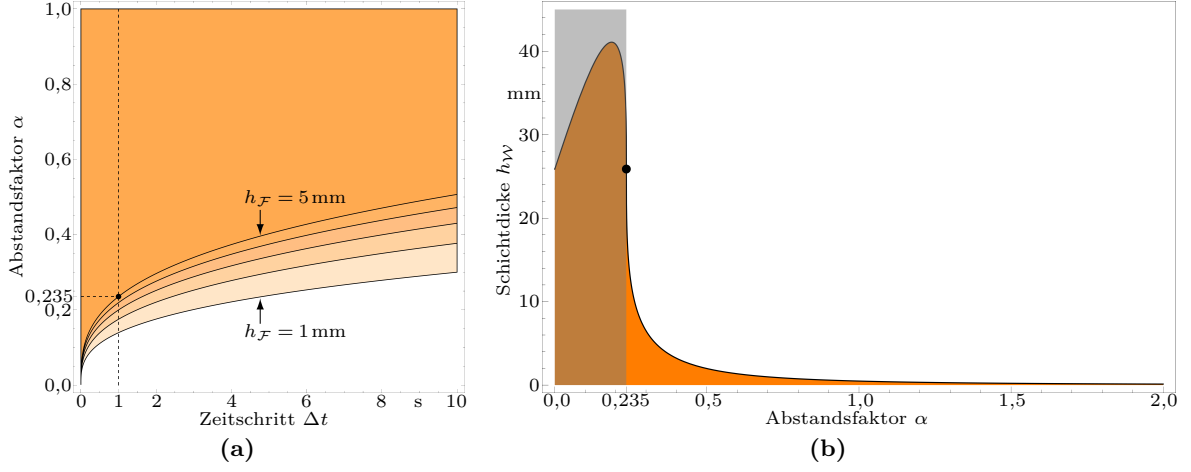


Abbildung 4.3: (a) Gültiger Parameterbereich (orange) der geometrischen Übertragungsfunktion für Footprintprofilstärken $h_{\mathcal{F}} = 1, \dots, 5$ mm bei $|a_z| = 110$ mm sowie $t_{\mathcal{F}} = 10$ s. (b) Verlauf der für den Fall $h_{\mathcal{F}} = 5$ mm von der geometrischen Übertragungsfunktion berechneten Schichtdicken $h_{\mathcal{W}}$ für $\Delta t = 1$ s (senkrechte gestrichelte Linie in Grafik (a)) in Abhängigkeit von der Entfernung des Brenners von der Werkstückoberfläche. Der Verlauf im schattierten Bereich ist nach Ungleichung (4.19) außerhalb des gültigen Bereichs.

Abbildung 4.2a muss daher die berechnete Schichtdicke $h_{\mathcal{W}}$ stets kleiner sein als der Abstand $-\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle$ der Tangentialebene des betrachteten Punktes auf der Werkstückoberfläche von der Brennerdüse beziehungsweise der Spitze des Spritzkegels. Somit ergibt sich die Bedingung

$$\begin{aligned}
 -\alpha \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle &\geq \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle \left(\sqrt[3]{\left(\left(1 + \frac{h_{\mathcal{F}}}{a_z} \right)^3 - 1 \right) k_t + \alpha^3 - \alpha} \right) \\
 \Leftrightarrow -\alpha &\leq \sqrt[3]{\left(\left(1 + \frac{h_{\mathcal{F}}}{a_z} \right)^3 - 1 \right) k_t + \alpha^3 - \alpha} \\
 \Leftrightarrow 0 &\leq \left(\left(1 + \frac{h_{\mathcal{F}}}{a_z} \right)^3 - 1 \right) k_t + \alpha^3 \\
 \Leftrightarrow \alpha &\geq \sqrt[3]{\left(1 - \left(1 + \frac{h_{\mathcal{F}}}{a_z} \right)^3 \right) \frac{\Delta t}{t_{\mathcal{F}}}}. \tag{4.19}
 \end{aligned}$$

Im Rahmen einer Simulation sind $h_{\mathcal{F}}$, a_z und $t_{\mathcal{F}}$ durch den zugrunde liegenden Footprintversuch vorgegeben. In Übereinstimmung mit der geometrischen Interpretation muss der Abstandsfaktor α groß und die Spritzdauer Δt klein sein, um diese Ungleichung zu erfüllen. Abbildung 4.3a zeigt den orange gefärbten, gültigen Bereich in Abhängigkeit von Δt und α für exemplarische Footprintschichtdicken von $h_{\mathcal{F}} = 1, \dots, 5$ mm bei einer Footprintspritzdauer von $t_{\mathcal{F}} = 10$ s und dem Standardabstand $|a_z| = 110$ mm des verwendeten Lichtbogenspritzprozesses. Je größer die Footprintschichtdicke oder das Simulationsintervall Δt , desto mehr Material wird abgelagert und desto größer muss der Abstand $\alpha \cdot 110$ mm des Brenners von der Werkstückoberfläche sein. Abbildung 4.3b zeigt den Verlauf von $h_{\mathcal{W}}$ entlang der gestrichelten,

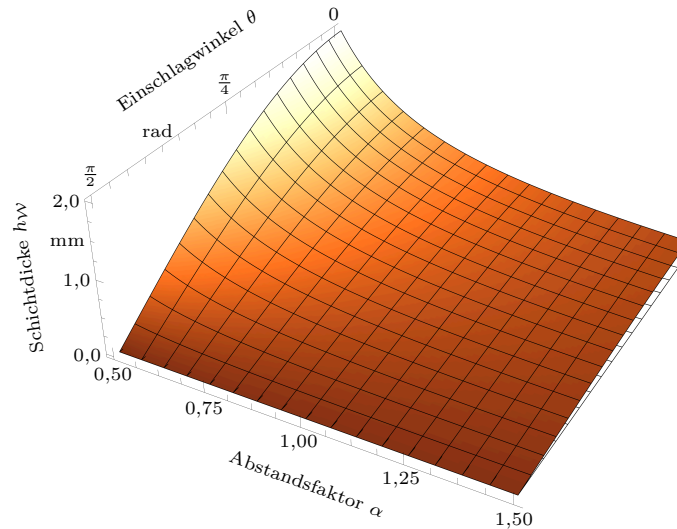


Abbildung 4.4: Abhängigkeit der geometrischen Übertragungsfunktion nach Gleichung (4.18) von Spritzabstand $\alpha \cdot 110\text{mm}$ und Einschlagwinkel θ ($t_{\mathcal{F}} = 10\text{s}$, $h_{\mathcal{F}} = 5\text{mm}$, $|a_z| = 110\text{mm}$).

vertikalen Linie in Abbildung 4.3a für $h_{\mathcal{W}} = 5\text{mm}$ bis zu einer Entfernung von 220 mm. Es ist hier ersichtlich, dass die Funktion im ungültigen Bereich (grau) in der Tat keinen sinnvollen Verlauf besitzt. Oberhalb von $\alpha = 0,235$ fällt die Funktion wie erwartet zunächst steil ab und konvergiert dann gegen Null. Für $\alpha = 1,0$ beträgt $h_{\mathcal{W}}$ beispielsweise 0,479 mm, also etwas weniger als $h_{\mathcal{F}}/10$, da nach $\frac{1}{10}$ -tel der Footprintspritzdauer zwar $\frac{1}{10}$ -tel des Volumens $dV_{\mathcal{F}}$ abgelagert wurde, aber die Schichthöhe im unteren Teil des spitz zulaufenden Kegelstumpfelements $dV_{\mathcal{W}}$ zunächst nur langsam wächst.

Abbildung 4.4 stellt die Abhängigkeit von $h_{\mathcal{W}}$ vom Entfernungsfaktor α und dem Einschlagwinkel θ dar und die Abbildungen 4.5a und 4.5b zeigen zur besseren Veranschaulichung des Einflusses der einzelnen Parameter einige Schnitte durch diese Darstellung. Gut erkennbar sind hier der erwartungsgemäß hyperlineare Anstieg der Schichtdicke mit sinkendem Spritzabstand sowie das kosinusförmige Ausblenden der Schichtdicke mit zunehmendem Einschlagwinkel.

4.3.3 Vereinfachte kegelbasierte Übertragungsfunktion

Während des durch die zuvor hergeleitete geometrische Übertragungsfunktion (4.18) beschriebenen Ablagerungsvorgangs innerhalb eines Zeitintervalls Δt wächst die Schicht bereits der Volumenstromquelle entgegen und der initiale Strahlquerschnitt $dA_{\mathcal{W}}$, auf dem neues Material abgelagert wird, verringert sich. Folglich steigt die Wachstumsgeschwindigkeit der Schicht bei gleichbleibendem Volumenstrom immer schneller an. Das abgelagerte Volumen $dV_{\mathcal{W}}$ wird als Kegelstumpf berechnet. Vernachlässigt man das Absinken des Querschnitts, so ist dies äquivalent dazu, die Volumina $dA_{\mathcal{F}}$ und $dA_{\mathcal{W}}$ als geraden Zylinder mit elliptischer Grundfläche aufzufassen. Ersetzt man dementsprechend Gleichungen (4.13) und (4.15) durch

$$dV_{\mathcal{F}} = h_{\mathcal{F}} dA_{\mathcal{F}} \quad \text{sowie} \quad dV_{\mathcal{W}} = \tilde{h}_{\mathcal{W}} dA_{\mathcal{W}},$$

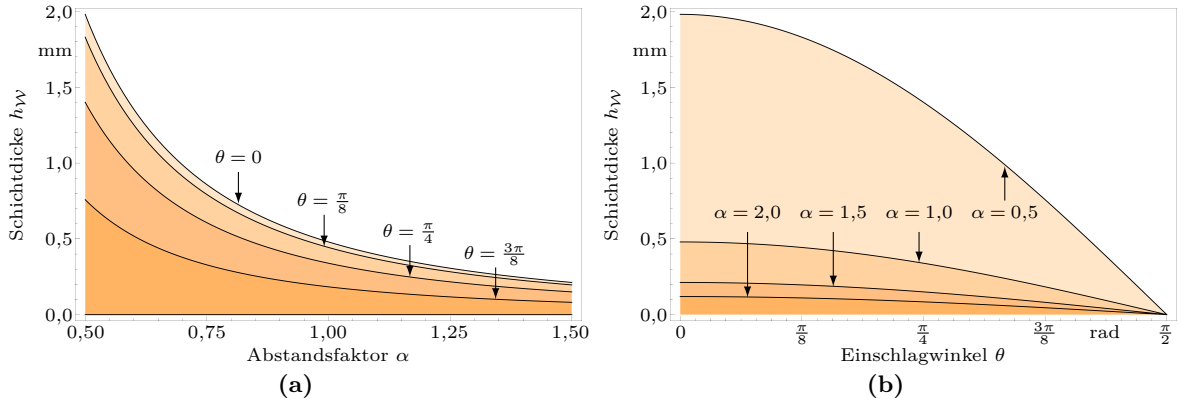


Abbildung 4.5: Mit der geometrischen Übertragungsfunktion (4.18) berechnete Schichtdicken auf dem Werkstück ($t_{\mathcal{F}} = 10\text{s}$, $h_{\mathcal{F}} = 5\text{ mm}$, $|a_z| = 110\text{ mm}$) dargestellt über (a) dem Abstandsfaktor α und (b) dem Einschlagwinkel θ .

so ergibt sich die approximative Übertragungsfunktion

$$\tilde{h}_{\mathcal{W}} = \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle \frac{k_t}{\alpha^2 a_z} h_{\mathcal{F}} = \frac{\cos(\theta_{\mathcal{W}})}{\cos(\theta_{\mathcal{F}})} \cdot \frac{1}{\alpha^2} \cdot \frac{\Delta t}{t_{\mathcal{F}}} \cdot h_{\mathcal{F}}. \quad (4.20)$$

Diese vereinfachte Gleichung ist nun leicht interpretierbar als Übertragung der Footprint- auf die Werkstückgeometrie. Zur Veranschaulichung werde ein ebenes Flächenelement dA , welches unter dem Winkel θ zur Ausbreitungsrichtung in einer homogenen und stationären Volumenströmung der Geschwindigkeit \mathbf{v} platziert ist, betrachtet. Nach Gleichung (4.7) ergibt sich hierfür ganz allgemein das in der Zeit dt durchströmende Volumen $dV_A = \cos(\theta) \cdot |\mathbf{v}| \cdot dA \cdot dt$. Für die Höhe des abgelagerten Volumens ergibt sich, weiterhin unter Vernachlässigung des sinkenden Strahlquerschnitts,

$$h = \frac{dV_A}{dA} = \cos(\theta) \cdot |\mathbf{v}| \cdot dt \quad (4.21)$$

beziehungsweise am Werkstück

$$h_{\mathcal{W}} = \cos(\theta_{\mathcal{W}}) \cdot |\mathbf{v}_{\mathcal{W}}| \cdot \Delta t. \quad (4.22)$$

Durch Umstellen von Gleichung (4.21) nach $|\mathbf{v}|$ ergibt sich wiederum der Betrag der Volumenstromdichte am Footprint

$$|\mathbf{v}_{\mathcal{F}}| = \frac{h_{\mathcal{F}}}{\cos(\theta_{\mathcal{F}}) t_{\mathcal{F}}}, \quad (4.23)$$

wobei die Footprintspritzdauer $t_{\mathcal{F}}$ die Zeit der Ablagerung auf dem Footprint ist. Aufgrund der Strahldivergenz, welche an dieser Stelle zwischen Footprint und Werkstück nicht vernachlässigt wird, ist der Betrag der Volumenstromdichte am Werkstück um den Faktor $1/\alpha^2$ kleiner als am Footprint. Mit den Gleichungen (4.22) und (4.23) ergibt sich daher

$$h_{\mathcal{W}} = \cos(\theta_{\mathcal{W}}) \cdot |\mathbf{v}_{\mathcal{W}}| \cdot \Delta t = \cos(\theta_{\mathcal{W}}) \cdot \frac{1}{\alpha^2} |\mathbf{v}_{\mathcal{F}}| \cdot \Delta t = \frac{\cos(\theta_{\mathcal{W}})}{\cos(\theta_{\mathcal{F}})} \cdot \frac{1}{\alpha^2} \cdot \frac{\Delta t}{t_{\mathcal{F}}} \cdot h_{\mathcal{F}},$$

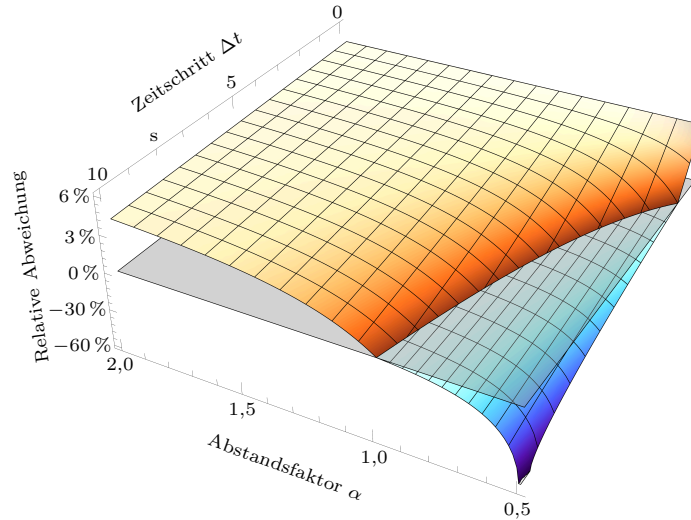


Abbildung 4.6: Prozentuale Über- beziehungsweise Unterschätzung $(\tilde{h}_{\mathcal{W}} - h_{\mathcal{W}})/h_{\mathcal{W}} \cdot 100\%$ der Schichtdicke durch die vereinfachte, kegelbasierte Übertragungsfunktion nach Gleichung (4.20). Für eine bessere Lesbarkeit ist die senkrechte Abweichungsachse im negativen Bereich um den Faktor 10 komprimiert.

also die approximative Übertragungsfunktion $\tilde{h}_{\mathcal{W}}$. Diese vereinfachte Übertragungsfunktion kann demzufolge als Verknüpfung einer invertierten Ablagerungsberechnung am Footprint mit einer nicht-invertierten Ablagerungsberechnung am Werkstück verstanden werden. Die vollständige Symmetrie zwischen footprint- und werkstückseitigen Termen wird durch die Ersetzung $1/\alpha^2 = (|\mathbf{a}|/|\alpha\mathbf{a}|)^2$ noch deutlicher. Es ist dann nach Gleichung (4.20):

$$\frac{\text{Höhe } \mathcal{W}}{\text{Höhe } \mathcal{F}} = \frac{\text{Neigungsterm } \mathcal{W}}{\text{Neigungsterm } \mathcal{F}} \cdot \left(\frac{\text{Distanz } \mathcal{F}}{\text{Distanz } \mathcal{W}} \right)^2 \cdot \frac{\text{Beschichtungsdauer } \mathcal{W}}{\text{Beschichtungsdauer } \mathcal{F}}.$$

Diese vereinfachte Modellierung auf Basis von Neigungs- und quadrierten Distanztermen weist Ähnlichkeiten mit der Berechnung der durch Bestrahlung einer Detektorfläche eingebrachten Energie in der Strahlungsphysik [125] oder auch Beleuchtungs- und Formfaktorberechnungen in der Computergrafik auf [30]. Für die im vorherigen Abschnitt ermittelte exakte Formulierung des Schichtwachstums gilt dies jedoch nicht, da der Querschnitt einer bestrahlten Detektorfläche im Gegensatz zur vorliegenden Materialablagerung nicht mit der eingetragenen Energie absinkt. Existierende Formulierungen aus dem Gebiet der Strahlungsphysik oder der Optik sind hier daher nicht direkt übertragbar.

Approximationsfehler

Im Folgenden wird nun der Fehler der Approximation nach Gleichung (4.20) gegenüber der exakten Formulierung aus Gleichung (4.18) betrachtet. Abbildung 4.6 zeigt hierzu den relativen Fehler $(\tilde{h}_{\mathcal{W}} - h_{\mathcal{W}})/h_{\mathcal{W}} \cdot 100\%$ in Abhängigkeit vom Entfernungsfaktor α und dem Zeitschritt Δt . Wie zuvor werden dabei weiterhin die für das Lichtbogenspritzen typischen Werte $h_{\mathcal{F}} = 5 \text{ mm}$, $|a_z| = 110 \text{ mm}$ und $t_{\mathcal{F}} = 10 \text{ s}$ angenommen. Sowohl für $\alpha \rightarrow \infty$ als auch für $\Delta t \rightarrow 0$

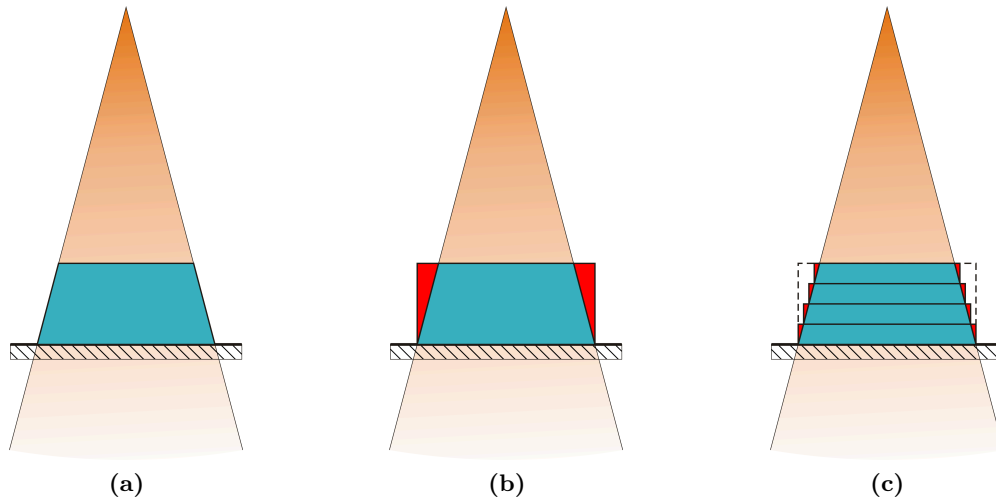


Abbildung 4.7: Vergleich des approximativen mit dem exakten Beschichtungsmodell. (a) Abgelegertes Kegelstumpfvolumen des exakten Modells. (b) Fehlervolumen (rot) bei Berechnung des Ablagerungsvolumens als Zylinder. (c) Reduzierter Fehler bei Ablagerungsberechnung in viermal so vielen Zeitschritten. Für die Verringerung des Fehlers muss in jedem Berechnungsschritt die Distanz zur Werkstückoberfläche unter Berücksichtigung der bereits abgelagerten Schichtdicke neu berechnet werden.

ist der prozentuale Fehler durch dieselbe Funktion beschränkt, welche nur noch vom Verhältnis $h_{\mathcal{F}}/d_{\mathcal{F}}$ abhängig ist. Die Unabhängigkeit dieser oberen Schranke von der Entfernung der Werkstückoberfläche ist geometrisch darin begründet, dass das Kegelstumpf-förmige Volumenelement $dV_{\mathcal{W}}$ mit zunehmender Entfernung tatsächlich in einen geraden Zylinder übergeht. Für den gewählten Spritzabstand des Footprints ist die Funktion im relevanten Bereich für $h_{\mathcal{F}}$ bis 10 mm näherungsweise linear und steigt um etwa einen Fehlerprozent pro Millimeter Profilstärke des Footprints. In dem in Abbildung 4.6 dargestellten Fall überschätzt die Approximation den exakten Schichtdickenwert um maximal 4,69%. Für größere Zeitschritte und geringere Abstände des Werkstücks von der Spritzpistole wird $h_{\mathcal{W}}$ hingegen unterschätzt, wobei der absolute prozentuale Fehler aufgrund der geringen Referenzschichtdicke stark ansteigt. Ebenfalls erkennbar ist, dass der Fehler im Fall $\Delta t = t_{\mathcal{F}} = 10\text{s}$ und $\alpha = 1$ verschwindet, da sich die Fehler bei der Approximation der Kegelstumpf-förmigen Volumenelemente durch Zylinder auf Seiten des Footprints und des Werkstücks aufheben.

Allgemein scheint bei diesen Fehlergrenzen eine Verwendung dieser Approximation zunächst ungeeignet, allerdings ist zu beachten, dass der hier berechnete prozentuale Fehler auf einen einzelnen Zeitschritt bezogen ist. Wird in jedem Zeitschritt der Simulation die bisher akkumulierte Schichtdicke bei der Berechnung des Spritzabstandes berücksichtigt, ist ein geringerer Zeitschritt insgesamt vorteilhaft und verursacht einen geringeren Fehler (Abbildung 4.7). Ebenso wird der Fehler dadurch beschränkt, dass der Spritzabstand des Footprints dem gewünschten und damit häufig auftretenden Arbeitsabstand entspricht, wodurch sich der durch die zylindrische Approximation verursachte Fehler auf Footprint- und Werkstückseite für be-

stimmte Positionen des Brenners näherungsweise ausgleichen kann. Für die Simulation der Beschichtung des Matrizenbauteils MAT3D (vergleiche Anhang D) ergab sich beispielsweise ein durchschnittlicher prozentualer Fehler von 0,27 % ($\sigma = 0,06\%$).

Vereinfachung für einen schmalen Spritzstrahl

Eine weitere Vereinfachung von $\tilde{h}_{\mathcal{W}}$ ergibt sich aus der Approximation $\cos(\theta_{\mathcal{F}}) \approx 1$, welche insbesondere für einen sehr schmalen Spritzstrahl eine gute Näherung darstellt:

$$\tilde{\tilde{h}}_{\mathcal{W}} = \frac{\cos(\theta_{\mathcal{W}})}{1} \cdot \frac{1}{\alpha^2} \cdot \frac{\Delta t}{t_{\mathcal{F}}} \cdot h_{\mathcal{F}}. \quad (4.24)$$

Diese Approximation reduziert die berechneten Schichtdicken und kann damit auch im allgemeinen Fall der Überschätzung der Schichtdicke durch die erste Vereinfachung $\tilde{h}_{\mathcal{W}}$ in den praxisrelevanten Bereichen $\alpha \geq 1$ entgegen wirken. Für die Simulation des Bauteils MAT3D folgt daraus eine Verringerung des durchschnittlichen prozentualen Fehlers gegenüber der ersten Approximation $\tilde{h}_{\mathcal{W}}$ auf 0,10 % bei einer erhöhten Streuung $\sigma = 0,17\%$. In der Praxis kann daher $\tilde{\tilde{h}}_{\mathcal{W}}$ eine bessere Approximation darstellen als $\tilde{h}_{\mathcal{W}}$.

Betrachtet man die vorgestellten Approximationen unter dem Gesichtspunkt des notwendigen Berechnungsaufwandes, ist erkennbar, dass der Berechnungsaufwand für das exakte Modell, insbesondere aufgrund der enthaltenen Kubikwurzel, höher ist als der der Approximationen. Im Rahmen der Implementierung der Shaderprogramme für die Simulation (Kapitel 5) konnte jedoch, trotz Verwendung einer iterativen Kubikwurzel-Funktion für Werte des Typs `double`, keine Abnahme der Simulationsgeschwindigkeit bei Verwendung des exakten Modells festgestellt werden. Daher wurde für alle Simulationen im Rahmen dieser Arbeit das exakte Beschichtungsmodell verwendet.

4.3.4 Zylindrischer Spritzstrahl

Bei einigen Spritzverfahren, wie zum Beispiel dem HVOF-Spritzen, kann der Spritzstrahl eine zu vernachlässigende Divergenz aufweisen und daher als zylindrisch aufgefasst werden. Für die Modellierung der Übertragungsfunktion bedeutet dies, dass der Volumenfluss stets parallel und insgesamt senkrecht zur Footprintebene verläuft. Das Volumenelement $dV_{\mathcal{F}}$ am Footprint ist daher stets ein Kreiszyylinder und das Volumenelement am Werkstück, $dV_{\mathcal{W}}$, ein gerader Zylinder mit elliptischer Grundfläche. Die Übertragungsfunktion vereinfacht sich damit zu

$$h_{\mathcal{W}}^{\text{cyl}} = \left\langle \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}, \mathbf{n}_{\mathcal{W}} \right\rangle \cdot \frac{\Delta t}{t_{\mathcal{F}}} \cdot h_{\mathcal{F}} = -n_{\mathcal{W}z} \cdot \frac{\Delta t}{t_{\mathcal{F}}} \cdot h_{\mathcal{F}} = \cos(\theta_{\mathcal{W}}) \cdot \frac{\Delta t}{t_{\mathcal{F}}} \cdot h_{\mathcal{F}}, \quad (4.25)$$

wobei $(0, 0, -1)^T$ die Ausbreitungsrichtung des Strahls ist.

4.4 Haftwirkungsgrad

Die bisherigen Betrachtungen zur Modellierung des Massentransfers vom Brenner auf das Bauteil waren rein geometrischer Natur und berücksichtigten keine prozess- oder material-spezifischen Aspekte. Dabei wurde davon ausgegangen, dass das gesamte vom Brenner abgegebene Material auf der Werkstückoberfläche abgelagert wird. In der Praxis bleibt jedoch nur ein Teil der auf das Substrat bzw. die bereits entstandene Beschichtung auftreffenden Partikel haften und trägt zur finalen Beschichtung bei. Die restlichen Partikel prallen wieder von der Oberfläche ab. Der Haftwirkungsgrad ξ (engl.: sticking efficiency, deposition efficiency) bezeichnet dabei den Anteil des tatsächlich abgelagerten Materials am insgesamt auftreffenden Material. Bei dieser Betrachtung bleibt das Material, welches aufgrund der zeitweiligen Führung des Brenners außerhalb des zu beschichtenden Bauteils von vornherein am Bauteil vorbei emittiert wird, unberücksichtigt.

Bekannte Einflussgrößen auf die Hafteigenschaften von Partikeln auf der Werkstückoberfläche sind der Einschlagwinkel, die Geschwindigkeit, die Viskosität der Partikel beziehungsweise ihr Schmelzzustand sowie die Substrattemperatur. Ein steiler Einschlagwinkel, eine hohe Geschwindigkeit, schmelzflüssige Partikel und eine hohe Substrattemperatur begünstigen dabei eine gute Haftung und Verklammerung der Partikel und somit einen höheren Haftwirkungsgrad und eine höhere Auftragsrate [2, 85, 116].

Die Abnahme der Partikelhaftung bei flachen Einschlagwinkeln ist darauf zurückzuführen, dass die Partikel aufgrund ihrer vergleichsweise geringen Geschwindigkeitskomponente normal zur Oberfläche mit erhöhter Wahrscheinlichkeit abprallen [76]. Umgekehrt führen hohe normale Aufprallgeschwindigkeiten sowie auch vollständig geschmolzene Partikel ohne festen Kern zu einem besseren Fließen des Partikels in vorhandene Zwischenräume darunterliegender Schichten und somit zu einer besseren Verklammerung und Haftung. Demgegenüber stellt der Aufschlag eines (teil)erstarrten Partikels einen elastischeren Stoßvorgang dar, welcher mit erhöhter Wahrscheinlichkeit in einem Abprallen des Partikels resultiert. Das Fließen des Partikels wird des Weiteren durch Substrattemperaturen oberhalb einer material- und prozessabhängigen Transitionstemperatur begünstigt, welche die Abkühlung und Erstarrung des auftreffenden Partikels verzögern, bis die Ausbreitung und damit Verklammerung des Splats mit dem Untergrund abgeschlossen ist. Im Fall zu niedriger Substrattemperaturen kommt es durch eine zu schnelle Erstarrung der Ausbreitungsfront des Partikels zu einer Behinderung der Ausbreitung, wodurch ein Teil des flüssigen Materials von der Oberfläche zurückspritzt. Dieses Phänomen wird als „Splashing Impact“ bezeichnet [2].

In einer Vielzahl von Veröffentlichungen wurden experimentell gewonnene Ergebnisse zum Haftwirkungsgrad und zu Haftmechanismen für unterschiedliche thermische Spritzprozesse sowie für verschiedene Materialien untersucht [73, 79, 145, 156, 159]. In der Regel werden dabei zur Bestimmung des Haftwirkungsgrades Flachproben mittels einfacher Linear- oder Mäanderbahnen unter Variation eines oder mehrerer Parameter beschichtet und der Haftwirkungsgrad durch Messung des abgelagerten Volumens oder der abgelagerten Masse bestimmt. Im Folgenden werden zunächst die Ansätze zur Modellierung des Haftwirkungsgrades in an-

deren Beschichtungssimulationen sowie ausführlicher die im Rahmen dieser Arbeit entwickelte und implementierte Formulierung beschrieben. Im anschließenden Abschnitt wird auf die allgemeine Methodik zur experimentellen Bestimmung der Modellparameter für den Haftwirkungsgrad eingegangen, bevor die konkreten Messergebnisse und die resultierenden Parameter für den verwendeten Lichtbogenspritzprozess vorgestellt werden.

4.4.1 Modellierung des Haftwirkungsgrades

Für die Modellierung des Haftwirkungsgrades ξ im Rahmen der Berechnung des Schichtauftrags werden in der Literatur verschiedene Formulierungen verwendet, welche häufig ausschließlich oder zumindest primär vom Einschlagwinkel θ abhängig sind (vergleiche Abbildung 4.2). So verwenden beispielsweise Kang et al. zur Modellierung des Haftwirkungsgrades im Plasmaspritzprozess von θ abhängige Polynome zweiten Grades, welche auf Messungen der abgelagerten Masse unter in 10° -Schritten variierten Spritzwinkeln beruhen [76]. Einen vergleichbaren Ansatz basierend auf Polynomen in Abhängigkeit von θ nutzen auch Djuric und Grant, welche die zusätzlichen Bedingungen stellen, dass diese symmetrisch, an der Stelle Null gleich Eins und bis $\theta = 90^\circ$ monoton fallend sind [45]. In einfachen Simulationen, die sich bei der Beschichtung auf ebener Bauteile in konstantem Abstand und mit konstanter Brennerausrichtung beschränken, kann der Haftwirkungsgrad auch durch einen konstanten Faktor realisiert werden [15].

Im verwandten Gebiet des Sprühkompaktierens modellieren Mi und Grant den Haftwirkungsgrad mittels einer gewichteten Kosinusfunktion in Abhängigkeit von θ [103]. Eine auf diesem Gebiet häufig anzutreffende Formulierung, welche neben dem Einschlagwinkel auch den Aufschmelzgrad der Partikel berücksichtigt, ist die von Mathur et al. [100]. Eine Kalibrierung des temperaturabhängigen Teils dieser Formulierung scheint jedoch so komplex, dass dieser häufig geschätzt oder vollständig unterschlagen wird [34, 103].

Integration in die Übertragungsfunktion

Im Rahmen der entwickelten Simulation wird der Haftwirkungsgrad primär als Funktion des Spritzwinkels modelliert, welcher den wichtigsten Einflussfaktor für die untersuchten Schichtsysteme darstellt. Ferner kann eine indirekte Abhängigkeit des Wirkungsgrades vom Schmelzzustand der Partikel auf der hier betrachteten makroskopischen Ebene ohne die Simulation der thermischen Historie von Einzelpartikeln durch das Einbeziehen der Entfernung des Brenners von der Werkstückoberfläche berücksichtigt werden, da die Partikel mit zunehmendem Abstand vom Spritzbrenner erkalten und der Haftwirkungsgrad abnimmt. Da beide Parameterabhängigkeiten unabhängig voneinander modelliert werden können [100, 103], kann ein multiplikativer Ansatz

$$\xi(\theta, d) = \xi_{\text{ang}}(\theta) \cdot \xi_{\text{dist}}(d), \quad (4.26)$$

für den Gesamthaftwirkungsgrad gewählt werden, wobei $d = \alpha|\mathbf{a}|$ die Distanz zwischen Brenner und Werkstückoberfläche gemäß Abbildung 4.2a ist. Da der Haftwirkungsgrad den Anteil

des Materials, der zur Schichtbildung beiträgt, angibt, kann $\xi(\theta, d)$, ebenso wie der Zeitfaktor k_t , als Gewichtung für das berechnete infinitesimale Ablagerungsvolumen in die geometrische Übertragungsfunktion integriert werden. In der Herleitung der Übertragungsfunktion wird hierfür Gleichung (4.17) durch

$$dV_{\mathcal{W}} = \xi(\theta, d) \cdot \frac{\Delta t}{t_{\mathcal{F}}} \cdot dV_{\mathcal{F}} \quad (4.27)$$

ersetzt. Mit dieser Änderung ergibt sich für das Beschichtungsmodell aus Gleichung (4.18) die um den Haftwirkungsgrad erweiterte und nun vollständige Formulierung

$$h_{\mathcal{W}} = \langle \mathbf{a}, \mathbf{n}_{\mathcal{W}} \rangle \left(\sqrt[3]{\left(\left(1 + \frac{h_{\mathcal{F}}}{a_z} \right)^3 - 1 \right) k_t \cdot \xi(\theta, d) + \alpha^3 - \alpha} \right). \quad (4.28)$$

Analog ergibt sich für das auf einem zylindrischen Spritzstrahl basierende Modell nach Gleichung (4.25) nun

$$h_{\mathcal{W}}^{\text{cyl}} = -n_{\mathcal{W}_z} \cdot k_t \cdot \xi(\theta, d) \cdot h_{\mathcal{F}}. \quad (4.29)$$

Modellierung der Teilfunktionen

Zur Bestimmung der winkel- und distanzabhängigen Teilfunktionen $\xi_{\text{ang}}(\theta)$ und $\xi_{\text{dist}}(d)$ aus Gleichung (4.26) werden in der Regel Versuchsreihen durchgeführt, bei der gleichgroße planare Proben mittels einfacher planparalleler Mäanderbahnen unter verschiedenen Winkeln θ_i zur Spritzrichtung respektive verschiedenen Entfernungen d_j vom Brenner beschichtet und das abgelagerte Volumen oder die abgelagerte Masse gemessen wird. An dieser Stelle hat die Vermessung des Volumens bzw. der auf den Proben abgelagerten Schichtdicken gegenüber einer Massenbestimmung den Vorteil, dass eine mit dem Spritzwinkel oder -abstand variierende Porosität erfasst wird. Auf separate Terme zur expliziten Vorhersage einer winkel- oder distanzabhängigen Porosität und einer resultierenden Steigerung der Schichtdicke kann daher bei der Formulierung des Beschichtungsmodells verzichtet werden.

Um beliebige Prozessparameterabhängigkeiten auf Basis weniger Messpunkte auf einfache Weise in die Simulation übernehmen zu können, werden die Funktionen $\xi_{\text{ang}}(\theta)$ und $\xi_{\text{dist}}(d)$ mittels kubischer Hermiteinterpolation auf Basis von Stützstellen $(\theta_i, \xi_{\text{ang},i})$ respektive $(d_j, \xi_{\text{dist},j})$ modelliert [58]. Die Notwendigkeit, für jeden Prozess oder jede Prozessparametrisierung neue Fittingpolynome zu berechnen und zu implementieren wird so vermieden. Darüber hinaus ist die gewonnene lokale Kontrollierbarkeit der Hermite-Splineinterpolation vorteilhaft für die automatische Parameterbestimmung, welche in Kapitel 5.8.1 beschrieben wird.

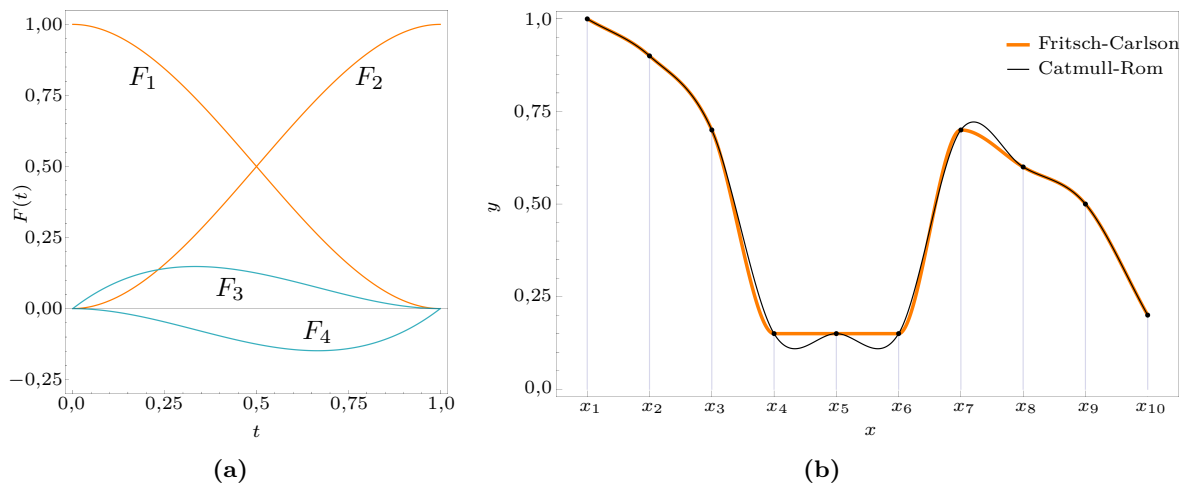


Abbildung 4.8: (a) Basisfunktionen F_1 bis F_4 der Hermiteinterpolation. (b) Beispiel zur monotonen Hermiteinterpolation nach der Methode von Fritsch und Carlson [58], welche Überschwinger unterdrückt. Als Vergleich dient hier die Catmull-Rom-Hermiteinterpolation [22].

Bei der Hermiteinterpolation wird eine glatte, auch in den Stützstellen stetig differenzierbare Interpolierende erzeugt, indem benachbarte Stützstellen mit kubischen Polynomkurven verbunden werden, die nicht nur die Koordinaten, im Folgenden allgemein mit (x_i, y_i) bezeichnet, sondern auch an den Stützstellen vorgegebene Steigungen m_i interpolieren [123]. Die kubische Hermite-Interpolationsfunktion $p(x)$ basiert auf den vier Hermite-Basisfunktionen

$$\begin{aligned} F_1(t) &= 2t^3 - 3t^2 + 1 \\ F_2(t) &= -2t^3 + 3t^2 \\ F_3(t) &= t^3 - 2t^2 + t \\ F_4(t) &= t^3 - t^2 \end{aligned} \quad , t \in [0, 1] \quad (4.30)$$

und lautet auf dem Intervall $[x_i, x_{i+1}]$

$$p(x) = y_i \cdot F_1(t) + y_{i+1} \cdot F_2(t) + h_i m_i \cdot F_3(t) + h_i m_{i+1} \cdot F_4(t),$$

wobei

$$t = \frac{x - x_i}{h_i} \quad \text{und} \quad h_i = x_{i+1} - x_i.$$

In Matrix-Vektor-Schreibweise lässt sich dies kompakt als

$$p(x) = \begin{bmatrix} x^3 & x^2 & x & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_i \\ y_{i+1} \\ h_i m_i \\ h_i m_{i+1} \end{bmatrix}$$

darstellen. Anschaulich interpolieren die ersten beiden (F_1 und F_2) der vier in Abbildung 4.8a dargestellten Basisfunktionen die Koordinaten und die letzten beiden (F_3 und F_4) die Steigungen.

Während die zu interpolierenden Punkte (x_i, y_i) in der Regel, wie hier durch die Messwerte des Haftwirkungsgrades, gegeben sind, existieren verschiedene Verfahren zur automatischen Berechnung der benötigten Steigungen für eine Folge von Stützstellen, welche üblicherweise auf den Sekantensteigungen der an x_i angrenzenden Intervalle $[x_{i-1}, x_i]$ und $[x_i, x_{i+1}]$ oder des übergreifenden Intervalls $[x_{i-1}, x_{i+1}]$ sowie weiteren Formparametern beruhen. Für die Auswahl eines geeigneten Verfahrens wurde an dieser Stelle als Kriterium die für die Interpolation technischer Kennwerte wichtige Vermeidung von Überschwingern der Interpolierenden herangezogen. Das bedeutet: $p(x)$ soll für monoton fallende (steigende) Stützstellen ebenfalls monoton fallend (steigend) sein; insbesondere soll $p(x)$ im Fall $y_i = y_{i+1}$ auf dem Intervall konstant sein. Dies wird durch die Implementierung des mehrschrittigen Verfahrens von Fritsch und Carlson [58] zur Bestimmung und Beschränkung der Steigungen m_i sichergestellt. Abbildung 4.8b zeigt beispielhaft eine Hermiteinterpolation nach Fritsch und Carlson im Vergleich mit dem Verfahren nach Catmull und Rom [22], welches nicht monotonieerhaltend ist.

4.4.2 Experimentelle Bestimmung des Haftwirkungsgrades

Geeignete Stützstellen $(\theta_i, \xi_{\text{ang},i})$ und $(d_j, \xi_{\text{dist},j})$ für die Modellierung des Haftwirkungsgrades werden mittels einfacher Spritzexperimente ermittelt. Ausgehend von einer gemeinsamen Parameterkonfiguration (θ^*, d^*) werden zu diesem Zweck mehrere planare Proben mittels planparalleler Mäanderbahnen beschichtet (Abbildung 4.9), wobei einmal der Spitzwinkel θ und einmal der Spritzabstand d in mehreren Schritten θ_i ($i = 1, \dots, k$) und d_j ($j = 1, \dots, l$) unabhängig voneinander variiert werden. Die gemeinsame Konfiguration (θ^*, d^*) entspricht dabei sinnvollerweise der Standardkonfiguration des Prozesses und dient im Rahmen der Definition des Haftwirkungsgrades als Bezugskonfiguration, für die o.B.d.A. ein Wirkungsgrad von Eins, also

$$\xi_{\text{ang}}(\theta^*) = \xi_{\text{dist}}(d^*) = \xi(\theta^*, d^*) = 1,$$

angenommen wird². Es seien im Folgenden $h(\theta_i)$ und $h(d_j)$ die über die Probenoberfläche gemittelten, im Experiment erzielten Schichtdicken für die jeweiligen Winkel θ_i respektive Abstände d_j sowie $h^* = h(\theta^*) = h(d^*)$ die für die Bezugskonfiguration gemessene Schichtdicke. Um aussagekräftige Messwerte zu erhalten, ist bei der Durchführung der Versuche darauf zu achten, dass die von den Bahnpunkten der Mäanderbahn in der Probenebene überdeckte Fläche groß genug ist, so dass sich die Probe am Beginn und Ende der Bahn noch nicht bzw. nicht mehr im Eingriff befindet. Dies kann durch die Wahl der folgenden in Abbildung 4.10 dargestellten Abstände sichergestellt werden:

$$\begin{aligned} d_v &= d_j \sin \gamma / \sin(90 - \theta_i - \gamma) \\ d_n &= d_j \sin \gamma / \sin(90 + \theta_i - \gamma). \end{aligned}$$

²Sollte für eine andere Parametrisierung ein höherer Wirkungsgrad erzielt werden, so können auch mit Werten $\xi > 1$ Simulationen durchgeführt werden.

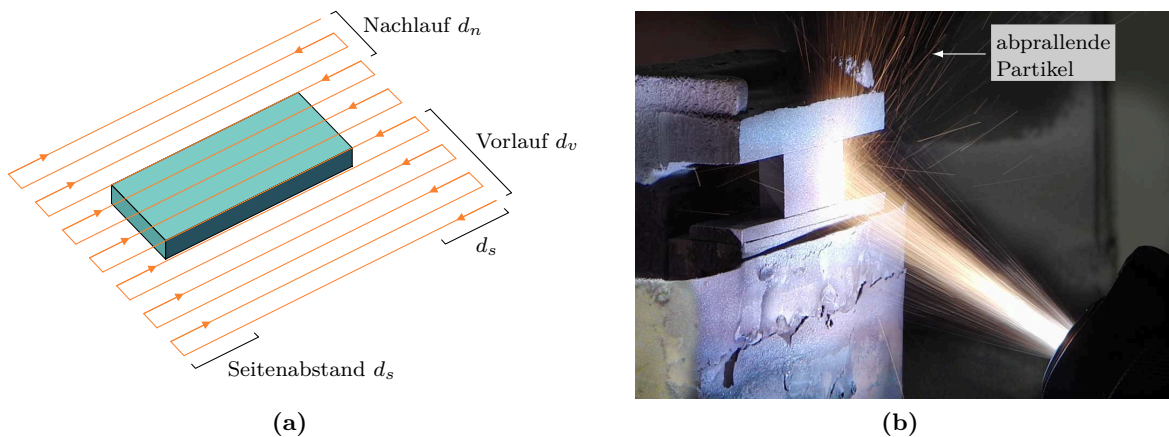


Abbildung 4.9: (a) TCP-Mäanderbahnverlauf bei der Beschichtung von Flachproben zur Bestimmung des Haftwirkungsgrades mit Vorlaufabstand, Nachlaufabstand sowie Seitenabständen. (b) Beschichtung einer Flachprobe unter dem Spritzwinkel 30° .

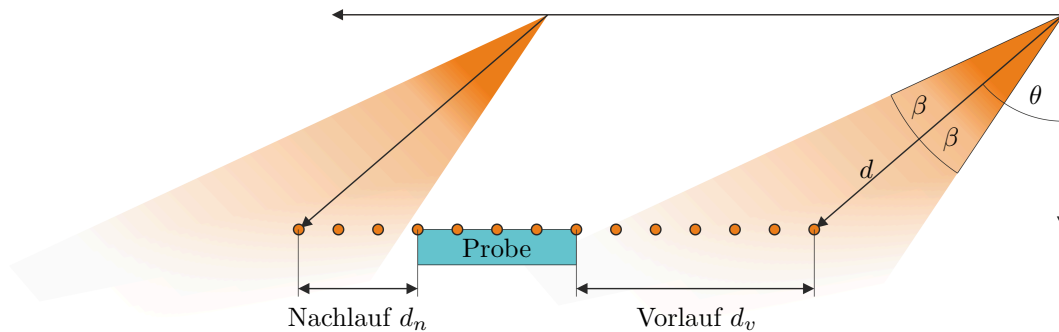


Abbildung 4.10: Vor- und Nachlaufabstände d_v und d_n für die Mäanderbahn aus Abbildung 4.9a bei Spritzwinkel θ , Stahlhalbwinkel β und Spritzabstand d .

Darin bezeichnet γ den halben Öffnungswinkel des Spritzstrahls. Um ferner zu gewährleisten, dass der Roboter den Brenner auf die gewünschte Geschwindigkeit beschleunigt hat, bevor der Spritzstrahl die Probe erreicht, müssen die seitlichen Umkehrpunkte der Bahn weit genug vom Probenrand entfernt liegen (Abbildung 4.9a).

Für einen hinreichend klein gewählten Bahnabstand ist so sichergestellt, dass jeder Punkt der Probenoberfläche in seiner Relativbewegung zum Brenner in guter Näherung den gesamten durch die Probenebene definierten Spritzstrahlquerschnitt mit konstanter Geschwindigkeit durchläuft. Da der vom Brenner im stationären Prozess abgegebene Massenstrom konstant ist, resultiert in diesem Fall, abgesehen von einer vernachlässigbaren Welligkeit aufgrund des Bahnabstandes, eine gleichmäßige Schichtdicke auf der gesamten Probe. Für einen theoretischen Haftwirkungsgrades $\xi(\theta, d) = 1$ würde insbesondere für beliebige Spritzwinkel und -entfernungen stets die gleiche Schichtdicke resultieren, so dass Abweichungen von der Schichtdicke der Bezugskonfiguration h^* dem Haftwirkungsgrad zugeschrieben werden können.

Unter der vereinfachenden Annahme, dass alle Punkte auf der Probenoberfläche während eines Versuchs stets unter dem exakt gleichen Winkel, aus konstanter Entfernung sowie für die

gleiche Gesamtdauer beschichtet werden, kann der Wirkungsgrad abgeschätzt werden als

$$\xi_{\text{ang},i} = h(\theta_i)/h^* \quad \text{bzw.} \quad \xi_{\text{dist},j} = h(d_j)/h^*. \quad (4.31)$$

Diese Annahme ist jedoch nur für einen unendlich schmalen, idealisierten Spritzstrahl gültig. In der Realität wird jeder Punkt auf der Oberfläche aufgrund des nicht zu vernachlässigenden Spritzstrahldurchmessers bei jedem Versuch unter verschiedenen Winkeln und Entfernungen beschichtet. Da ferner die Spritzrichtungsachse des in das Roboterverfahrensystem programmierten Werkzeugkoordinatensystems nicht unbedingt der durchschnittlichen Massenstromrichtung entsprechen muss, ist theoretisch nicht einmal sichergestellt, dass der Partikeleinschlagwinkel bei einem solchen Versuch im Durchschnitt mit dem programmierten Anstellwinkel übereinstimmt. Diese Probleme werden durch die entwickelte, automatische Kalibriermethodik umgangen.

4.4.3 Automatische Kalibrierung

Für eine genauere Bestimmung des Haftwirkungsgrades wurde das im Rahmen dieser Arbeit entwickelte Simulationsmodul zur automatischen Kalibrierung von Modellparametern verwendet, welches detailliert in Kapitel 5.8.1 beschrieben wird. Dabei werden die Stützstellen für die Teilfunktionen des Haftwirkungsgrades zunächst auf Basis der Messdaten nach Gleichung (4.31) initialisiert. Weiterhin wird jedes zur Bestimmung des Haftwirkungsgrades durchgeführte Experiment mittels der verwendeten Probengeometrie, Roboterbahn und des prozessspezifischen Footprints in dem entwickelten Simulationssystem modelliert und die gemessenen Schichtdicken werden als Zielfunktionswerte für die jeweiligen Simulationen vorgegeben. Anschließend werden die Stützstellenpositionen $\xi_{\text{ang},i}$ beziehungsweise $\xi_{\text{dist},j}$ iterativ und simultan für alle durchgeführten Versuche im Hinblick auf das Erreichen der vorgegebenen Zielwerte in der Simulation optimiert. Dabei wird zusätzlich zu den Messergebnissen für einen Beschichtungswinkel von $\theta = 90^\circ$ ein Haftwirkungsgrad von Null vorgegeben.

Für die Bezugsconfiguration (θ^*, d^*) wird im Rahmen dieser Parameteroptimierung ein Haftwirkungsgrad von Eins angenommen. Daher kann es durch Ungenauigkeiten bei der Footprint-erfassung oder der Messung von h^* vorkommen, dass für (θ^*, d^*) die gemessene Schichtdicke h^* simulativ nicht erreicht werden kann, da bei der Simulation der Flachprobenbeschichtung mit der Bezugsconfiguration kein freier Parameter zur Verfügung steht. Um dieses Problem zu umgehen, empfiehlt es sich, gleichzeitig mit dem Haftwirkungsgrad auch den Gesamtmassenstrom durch die Einführung eines Skalierungsparameters $s_{\mathcal{F}}$ für das Footprintprofil zu optimieren. Alternativ kann eine Anpassung des Gesamtmassenstroms auch im Vorhinein, beispielsweise mittels des in Kapitel 3.8 vorgestellten Ansatzes, durchgeführt werden.

Wird der Massenstrom simultan mit dem Haftwirkungsgrad optimiert und wurde für jede Stützstelle genau ein Versuch durchgeführt, werden für insgesamt n durchgeführte Messungen auch n Parameter optimiert. Für jeden Versuch mit Ausnahme der Bezugsconfiguration ist dies der Haftwirkungsgrad $\xi_{\text{ang},i}$ bzw. $\xi_{\text{dist},j}$ und für die Bezugsconfiguration der Faktor $s_{\mathcal{F}}$ für den Gesamtmassenstrom. Die Lösung ist in diesem Fall mathematisch exakt.

Mit dieser automatischen Kalibriermethode besteht die Möglichkeit, den Haftwirkungsgrad anhand gemessener Schichtdickenverteilungen auf komplexeren Bauteilen zu optimieren. Die hierfür verwendeten Bauteile und Beschichtungsbahnen sollten zu diesem Zweck alle relevanten Kombinationen von Spritzwinkeln und Distanzen in möglichst gleicher Häufigkeit aufweisen, um ein gutes Ergebnis durch die Kalibrierung zu erzielen. Vorteilhaft bei dieser Methodik ist die Reduktion der Anzahl durchzuführender Kalibrierversuche und die damit verbundene Zeit- und Kostenersparnis. Da die Fertigung und Vermessung komplexer Geometrien relativ aufwendig ist, kann ferner die Konstruktion einer segmentierbaren Kalibriergeometrie oder Probenaufspannung, in die einzelne Flachproben passgenau eingelegt werden können, vorteilhaft sein.

4.4.4 Haftwirkungsgrad für den verwendeten Lichtbogenspritzprozess

Für den im Rahmen dieser Arbeit verwendeten Lichtbogenspritzprozess wurde der Haftwirkungsgrad ausschließlich in Abhängigkeit des Spritzwinkels modelliert. Die Analyse der unter verschiedenen Distanzen angefertigten Footprintproben in Kapitel 3.9.3 deutete zwar auf eine Abhängigkeit des Gesamtmassenstroms von der Distanz hin, allerdings ist diese nur sehr gering ausgeprägt. Ausführlichere, von Krebs [85] dokumentierte Versuche zeigten für den vorliegenden Lichtbogenspritzprozess hohe Schwankungen in den Messungen und ein sehr geringes Bestimmtheitsmaß der Messungen. Für diesen Prozess wird daher $\xi_{\text{dist}}(d) = 1$ angenommen.

Zur Ermittlung der Winkelabhängigkeit wurden sieben Flachproben der Größe $50 \times 20 \text{ mm}^2$ mit programmierten Spritzwinkeln in 10° -Schritten im Bereich von $\theta = 0^\circ$ bis 60° beschichtet. Der Verlauf der Mäanderbahnen entspricht qualitativ der in Abbildung 4.9a gezeigten, wobei der Bahnabstand 5 mm und die Verfahrensgeschwindigkeit 200 mm/s betrug und für jede Probe vier Überläufe durchgeführt wurden. Die resultierenden Schichtdicken wurden anschließend entlang mehrerer Querschliffe lichtmikroskopisch aufgenommen und mit Hilfe des im Rahmen dieser Arbeit entwickelten und in Kapitel 3.7 vorgestellten Softwaretools ausgewertet. Die gemessenen Schichtdicken lagen im Bereich von 625 μm bis 738 μm mit Standardabweichungen zwischen 23 μm und 53 μm .

Mit Hilfe der automatischen Parameterkalibrierung wurde basierend auf den gemessenen initialen Stützstellen eine simultane Parameteroptimierung für die sechs Neigungswinkel $\theta_i = i \cdot 10^\circ$ für $i = 1, \dots, 6$ sowie für den Skalierungsfaktor $s_{\mathcal{F}}$ des Gesamtmassenstroms durchgeführt. Der Haftwirkungsgrad für $\theta = 90^\circ$ wurde auf Null gesetzt. Durch die Optimierung wurden die initialen Werte des Haftwirkungsgrades im Durchschnitt um 1,26 % und im Maximum um 3,1 % korrigiert. Die vorgegebenen Zielschichtdicken der Flachproben wurden, wie aufgrund der zur Verfügung stehenden Freiheitsgrade zu erwarten war, im Rahmen der Rechengenauigkeit simulativ exakt erreicht.

Der berechnete und in Abbildung 4.11 dargestellte Verlauf des spritzwinkelabhängigen Haftwirkungsgrades, welcher auch bei relativ flachen Spritzwinkeln bis 60° auf einem hohen Niveau bleibt, ist typisch für einen Lichtbogenspritzprozess, bei dem ein im Vergleich mit anderen

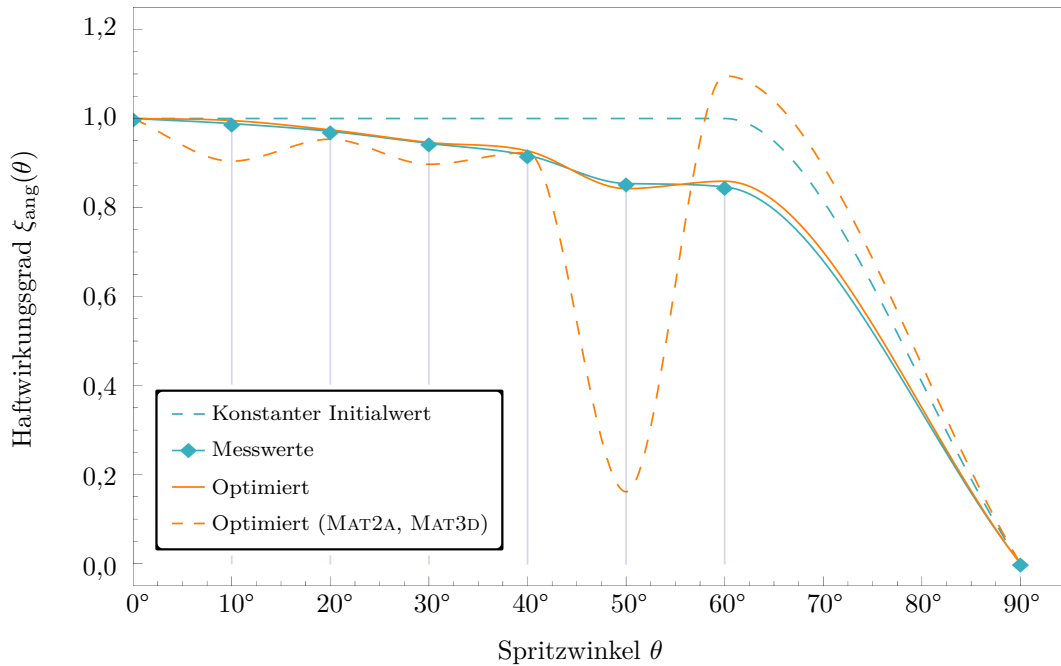


Abbildung 4.11: Unabhängig von der Initialisierung des Haftwirkungsgrades $\xi_{\text{ang}}(\theta)$ mit den gemessenen Werten oder einem konstanten Initialwert ergibt sich aus der Parameteroptimierung auf Basis der simulierten Flachprobenversuche ein den Messwerten ähnlicher Verlauf. Eine Optimierung basierend auf den komplexen Bauteilen MAT2A und MAT3D hingegen führt nur für die Stellen 0° , 20° und 40° zu ähnlichen Werten, da diese hinreichend oft bei der Beschichtung dieser Bauteile auftreten. Für Winkel $\theta > 50^\circ$ fehlen hingegen Informationen als Basis für eine erfolgreiche Kalibrierung. Ein spezielles Kalibrierbauteil wäre hier vorteilhaft.

Verfahren hoher Anteil des Spritzmaterials vollständig geschmolzen ist und somit auch unter flachen Winkeln nur zu keinen Teilen von der Substratoberfläche zurückprallt. Dies ist darauf zurückzuführen, dass sich nur vollständig geschmolzene Partikel als Tropfen vom Spritzdraht ablösen, wohingegen eingedüστε Pulverpartikel bei anderen Verfahren unter Umständen zu kurz erhitzt werden, um vollständig zu schmelzen.

Neben diesen Ergebnissen zeigt Abbildung 4.11 auch den stark schwankenden und daher nicht verwendbaren Haftwirkungsgrad nach einer Parameterkalibrierung basierend auf den Schichtdickenmessungen der Matrizenbauteile MAT2A und MAT3D anstelle der Flachproben. Die stärksten Abweichungen von der auf Basis der Flachprobenversuche optimierten Funktion treten hier für Winkel θ auf, die bei der Beschichtung der Matrizenbauteile nur an einem Bruchteil der Messpunkte vorkommen.

Beschichtungssimulation

Für die Simulation thermischer Spritzprozesse und die Berechnung der resultierenden Beschichtung sowie der weiteren bauteil- und bahnbezogenen Zielgrößen wurde im Rahmen dieser Arbeit ein umfangreiches Simulationsframework entwickelt, welches in diesem Kapitel vorgestellt wird. Abbildung 5.1 gibt einen Überblick über die Komponenten dieses Frameworks¹.

Den zentralen Teil stellen dabei die entworfenen GPU-basierten Algorithmen zur zeiteffizienten Simulation des Schichtwachstums mittels der im vorherigen Kapitel beschriebenen Beschichtungsmodelle dar, welche CPU-seitig in C++ und GPU-seitig in OpenGL (Open Graphics Library) beziehungsweise in Form von mehreren GLSL-Shaderprogrammen (OpenGL Shading Language) entworfen wurden, um von der im Vergleich zu CPUs deutlich höheren, parallelen Rechenleistung moderner Grafikkarten zu profitieren. Diese Kernalgorithmen sind in ein iteratives Zeitschrittverfahren eingebettet, in dem der virtuelle Spritzbrenner entlang einer gegebenen Roboterbahn verfahren wird.

Im folgenden Kapitel 5.1 werden zunächst die für das Verständnis der Implementierung notwendigen Grundlagen zu OpenGL und GLSL sowie zur Verwendung von GPUs für allgemeine Berechnungen beschrieben. Der übergreifende Simulationsablauf wird in Kapitel 5.2 detailliert beschrieben, um zunächst einen Überblick über das Simulationssystem zu vermitteln. Anschließend werden in Kapitel 5.3 die Eingabedaten und insbesondere die geometrische Repräsentation von Bauteil, Roboterbahn und Prozesskonfigurationsdaten dargestellt, bevor schließlich im Hauptteil 5.4 und 5.5 die GPU-basierten Simulationsalgorithmen sowie ihre Implementierung ausführlich beschrieben werden. Anschließend wird in den Kapiteln 5.6 und 5.7 das Interface zur Anbindung einer thermodynamischen Simulationskomponente sowie die Visualisierung der Simulationsergebnisse vorgestellt. Im letzten Abschnitt 5.8 wird auf die

¹Für eine Orientierung in diesem Kapitel können die Kapitelverweise in Abbildung 5.1 verwendet werden.

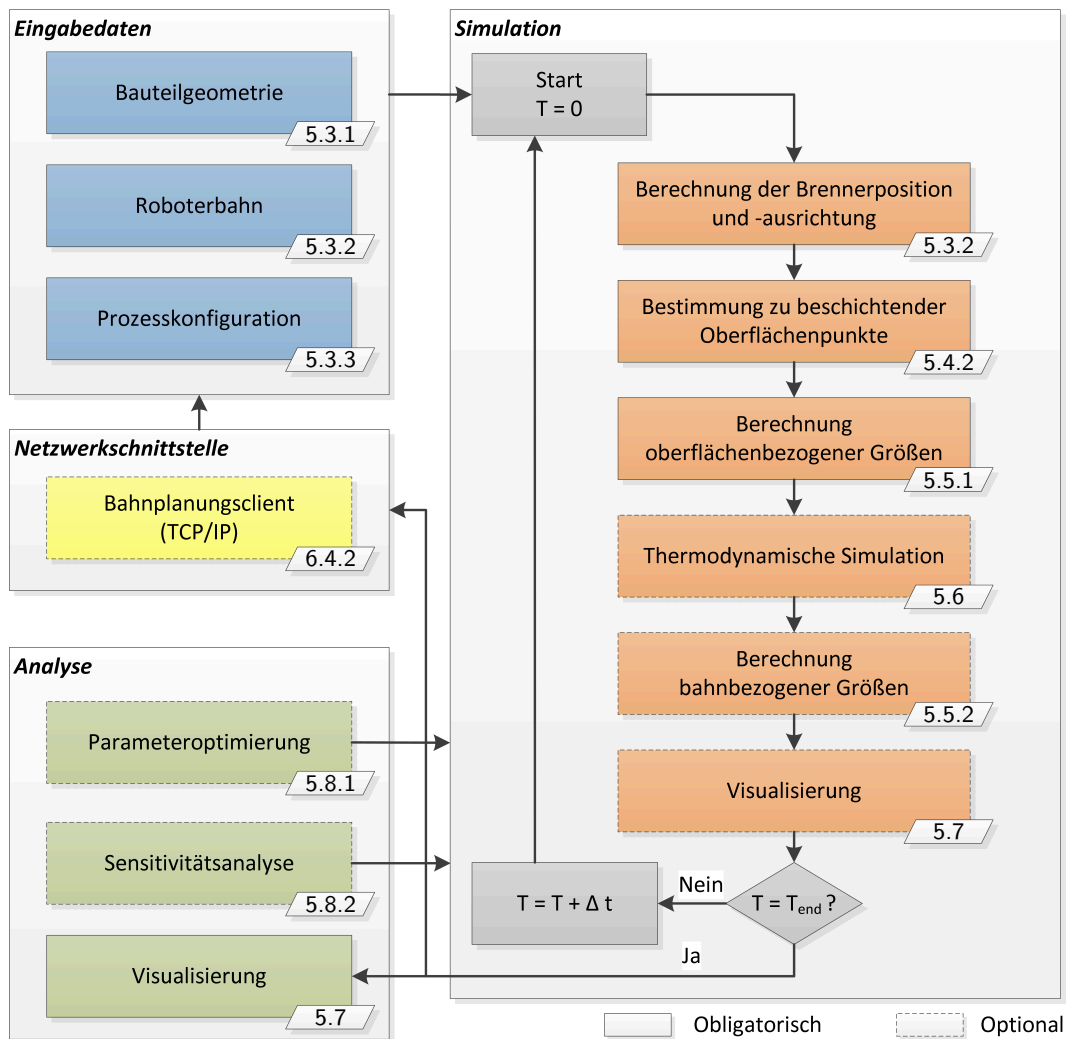


Abbildung 5.1: Übersicht über die Komponenten und den übergeordneten Ablauf der Simulation. Die Nummerierung gibt die zugehörigen Kapitel an, in denen die Komponente oder der Ablaufschritt beschrieben werden.

Implementierung zweier Module zur automatischen Kalibrierung von Simulationsparametern und zur Durchführung simulativer Sensitivitätsanalysen eingegangen, welche auf der Beschichtungssimulation aufbauen.

5.1 OpenGL, GLSL und GPGPU

Im folgenden Abschnitt 5.1.1 werden die Open Graphics Library (OpenGL) und die Shaderprogrammiersprache GLSL beschrieben. Die Ausführungen beschränken sich darin auf die für das Verständnis der Implementierung der Beschichtungssimulation notwendigen Aspekte. Das anschließende Kapitel 5.1.2 gibt einen Überblick über die Entwicklung von Methoden zur Nutzung von GPUs für allgemeine Berechnungen.

5.1.1 OpenGL und GLSL

Die OpenGL (Open Graphics Library) ist eine plattform- und programmiersprachenunabhängige Softwarebibliothek für das Rendern zwei- und dreidimensionaler Grafikszenen, welche ursprünglich von Silicon Graphics Inc. (SGI) entwickelt und 1992 in der Version 1.0 veröffentlicht wurde [130]. Um die Darstellung komplexer virtueller Szenen in Echtzeit zu erlauben, ist die Spezifikation darauf ausgelegt, in Hardware implementiert zu werden. Implementierungen der Spezifikation stellen typischerweise eine Schnittstelle zur softwaretechnischen Interaktion mit Grafikkarten dar und werden im Allgemeinen als Grafikkartentreiber oder kurz Grafiktreiber bezeichnet.

Die Funktionsweise von OpenGL kann als Grafikpipeline beschrieben werden, welche die einzelnen algorithmischen Schritte von der Eingabe der Geometriedaten einer virtuellen Szene bis zur Ausgabe der gerenderten Szene als Menge von Pixelfarbwerten im Framebuffer definiert. Dabei werden zunächst die darzustellenden Szenen als Menge von Koordinaten, welche als Vertices bezeichnet werden, mit zusätzlichen Attributen von der Anwendung definiert. Zur sequentiell ausgeführten *klassischen* OpenGL-Rendering-Pipeline gehören unter anderem die folgenden wichtigsten Schritte:

- **Transformation**

In diesem Schritt werden die eingegebenen Vertices, in der Regel durch Anwendung von Translationen, Rotationen und gegebenenfalls Skalierungen, in das Koordinatensystem der virtuellen Kamera transformiert.

- **Beleuchtungsberechnung**

Basierend auf definierten Lichtquellen, Farbwerten und Eigenschaften der Vertices sowie verschiedenen Beleuchtungsmodellparametern werden in diesem Schritt Farbwerte für die Vertices berechnet.

- **Primitive Assembly**

Die einzelnen Vertices werden in der „Primitive Assembly“ zu logischen geometrischen Primitiven, das heißt Dreiecken, Linien oder Punkten, zusammengefasst.

- **Clipping**

Durch die Definition der virtuellen Kamera unter Verwendung einer perspektivischen Projektion wird ein Pyramidenstumpf-förmiges Sichtvolumen definiert. Teile der Szene, die außerhalb dieses Sichtvolumens liegen, sind von der Kameraposition aus nicht sichtbar und werden im sogenannten „Clipping“-Schritt entfernt. Primitive, welche außerhalb des Sichtvolumens liegen, werden dabei vollständig entfernt. Primitive, die die Berandung des Sichtvolumens schneiden, werden auf das Sichtvolumen beschnitten.

- **Projektion**

Die aus dem vorherigen Schritt verbleibenden sichtbaren Primitive werden entsprechend der durch die Kamera definierten Projektionsvorschrift auf die virtuelle zweidimensionale Bildebene projiziert.

- **Verrasterung**

Im Rahmen der Verrasterung (engl.: rasterization) werden die nun in der zweidimensionalen Bildebene vorliegenden Primitive in eine Pixel-bezogene Darstellung konvertiert. Dabei wird für jede Pixelposition des finalen Framebuffers, welche von dem Primitiv überdeckt wird, ein sogenanntes „Fragment“ generiert, welches später zur Bestimmung eines Farbwertes an dieser Position im Framebuffer beitragen kann, falls es nicht von Fragmenten anderer Primitive an dieser Position verdeckt wird.

- **Schattierung (Shading)**

Entsprechend des gewählten Schattierungsmodells, typischerweise Flat-, Gouraud- oder Phong-Shading, werden Farbwerte für die Fragmente eines Primitivs basierend auf den Farbwerten oder Normalen seiner Vertices durch Interpolation berechnet.

- **Tiefenpuffertest**

In diesem Pipelineschritt werden Teile der Szene verworfen, die von anderen Teilen verdeckt werden und daher nicht sichtbar sind. Dies geschieht durch die Anwendung des Tiefenpufferverfahrens (engl.: depth buffer algorithm, z-buffer algorithm), welches nur die Farbwerte derjenigen Fragmente in die Ausgabe im Framebuffer übernimmt, welche an jeder Pixelposition der virtuellen Kamera am nächsten sind. Diese Berechnung basiert auf zuvor berechneten Tiefenwerten für jedes Fragment.

- **Darstellung**

Nachdem die gesamte geometrische Beschreibung der virtuellen Szene die Pipeline durchlaufen hat, stellen die im Framebuffer enthaltenen Farbwerte die für die Kameraposition berechnete zweidimensionale Abbildung der Szene dar, welche auf einem Bildschirm dargestellt werden kann.

Dieses grundlegende Vorgehen stellt die bis heute gültige Basis für die Funktionsweise von OpenGL dar. Seit der Einführung dieser Pipeline wurden jedoch viele Teilschritte und neue Funktionalitäten hinzugefügt. Ein Meilenstein in der Entwicklung von OpenGL stellte dabei die Einführung der OpenGL Shading Language (GLSL) mit der OpenGL Version 2.0 (2004) dar, mit der zwei der bisher starr definierten Einzelschritte der Pipeline mittels Vertex- und Fragmentshadern programmierbar wurden. Diese Entwicklung hat sich fortgesetzt und in den aktuellen, 2014 erschienenen OpenGL und GLSL Versionen 4.5 können frei programmierte Geometry-, Tessellation Control-, Tessellation Evaluation- und Compute-Shader Teile der Pipeline ersetzen oder ergänzen und dadurch unterschiedliche erweiterte Funktionalitäten realisieren. Die GLSL-Spezifikation [78] definiert zu diesem Zweck für die genannten Shadertypen unterschiedliche, jedoch sehr eng verwandte Programmiersprachen. Shaderprogramme, die in diesen Sprachen geschrieben wurden, können mittels OpenGL-API-Befehlen kompiliert, in die Grafikpipeline eingebunden und aktiviert werden. Die beiden wichtigsten im Rahmen dieser Arbeit verwendeten Shadertypen stellen Vertex- und Fragmentshader dar:

- **Vertexshader**

Vertexshaderprogramme ersetzen in der beschriebenen Pipeline die Schritte Transformation und Beleuchtungsberechnung, welche zusammengefasst auch als „Transform and Lighting“-, oder kurz T&L-Schritt bezeichnet wird. Ein Vertexshader bearbeitet stets einen einzelnen Vertex und hat dabei lesenden Zugriff auf die diesem Vertex zugeordneten Eingabedaten, darunter die Koordinaten des Vertex. Weitere typische Vertexdaten stellen Farbwerte und Normalenvektoren dar, prinzipiell können jedoch beliebige Daten für jeden Vertex definiert werden. Auf Basis dieser Eingaben sowie eventuell weiteren Daten, die aus dem Texturspeicher gelesen werden können, berechnet der Vertexshader die Ausgabe, welche insbesondere die Koordinaten `gl_Position` des transformierten Vertex enthält. Zusätzlich können beliebige weitere, Vertex-bezogene Ausgabedaten definiert werden, welche von anderen Shaderprogrammen im weiteren Verlauf der Pipeline verwendet werden können.

- **Fragmentsshader**

Der Fragmentshader ersetzt in der zuvor beschriebenen Pipeline den Schattierungsschritt. Entsprechend der Position in der Pipeline besteht die Eingabe eines Fragmentprogramms aus einem einzelnen Fragment und den ihm zugeordneten Daten aus früheren Schritten in der Pipeline. Im Rahmen der Programmausführung können weder die Daten anderer Fragmente gelesen, noch die Position des Fragments im Framebuffer verändert werden. Klassischerweise ist die Ausgabe eines Fragmentshaders die Farbe des Fragments, welche durch Anwendung von Beleuchtungs- oder Schattierungsalgorithmen berechnet wird. Mit der Einführung der Multiple-Render-Target (MRT) Techniken können auch aus einem Fragmentshader mehrere Datenwerte ausgegeben werden, welche von OpenGL in unterschiedlichen Framebuffers abgelegt werden.

Für eine detaillierte Beschreibung der Vertex- und Fragmentsprachen sowie der weiteren Shadertypen und ihrer Einbindung in das OpenGL-API wird auf die GLSL-Spezifikation [78], die OpenGL-Spezifikation [130] und die umfangreiche Fachliteratur [176] verwiesen.

In der Renderingpipeline werden die Daten von der Eingabe als Vertices bis zur Ausgabe im Framebuffer unidirektional propagiert. Dies bedeutet, dass Shaderprogramme im Rahmen der Renderingpipeline nicht auf Daten beliebiger Vertices, Primitive oder Fragmente zugreifen können und der Kontrollfluss zwischen den Einzelschritten nur in einer Richtung verläuft. Diese Einschränkungen ermöglichen die effiziente parallele Ausführung eines Schritts der Pipeline beziehungsweise eines Shaderprogramms für eine Vielzahl von Vertices oder Fragmente. Aufgrund dieser Gegebenheiten und der hohen Anzahl der im Echtzeit-Rendering in engen zeitlichen Grenzen zu verarbeitenden Datenelemente haben sich moderne Grafikkarten zu massiv parallelen Prozessorarchitekturen entwickelt. Ein einzelner Grafikprozessor (engl.: Graphics Processing Unit, GPU) weist dabei nach aktuellem Stand bis zu 3.072 einzelne Prozessorkerne auf², welche die parallele Ausführung von ebenso vielen Shaderprogrammen erlauben.

²Nvidia GeForce GTX Titan X, <http://nvidia.com>

Einen wichtigen Schritt stellte in diesem Zusammenhang die Einführung der Unified-Shader-Architektur (Shader Model 4.0) dar, mit der der Basisinstruktionssatz für die verschiedenen Shaderprogrammiersprachen vereinheitlicht und hardwareseitig nur noch ein Typ von Recheneinheiten benötigt wurde. Damit kann die Rechenleistung der zur Verfügung stehenden Einheiten bedarfsgerecht den einzelnen Shaderprogrammen zugeteilt und so die Auslastung der Hardware maximiert werden [105].

5.1.2 GPGPU

Der Begriff GPGPU (engl.: General-Purpose computation on Graphics Processing Units) beschreibt die Nutzung von Grafikprozessoren zur Durchführung allgemeiner, nicht Grafikbezogener Berechnungen. Gelingt es dabei, die durchzuführenden Berechnungen derart zu parallelisieren, dass die GPU in hohem Maße ausgelastet wird, können im Vergleich zu einer herkömmlichen, CPU-basierten Implementierung in der Regel deutliche Geschwindigkeitsvorteile erzielt werden. Dies ist wesentlich auf die im Vergleich zu herkömmlichen CPU-Architekturen, welche für die effiziente serielle Ausführung von Programmen optimiert wurden, deutlich höhere Gesamtrechenleistung paralleler GPU-Architekturen zurückzuführen. So liegt beispielsweise die theoretische Peak-Rechenleistung, gemessen in Milliarden Gleitkommaoperationen einfacher Genauigkeit pro Sekunde (engl.: Giga Floating point Operations Per Second, GFLOPS), der schnellsten momentan verfügbaren Desktop-GPU, der Nvidia GeForce GTX Titan X, bei 6.600 GFLOPS. Im Vergleich dazu erreicht die derzeit schnellste Desktop CPU (Intel Core-i7 5960X) mit 8 Prozessorkernen maximale 384 GFLOPS. Im professionellen Workstation Markt erreicht die Intel Xeon E7-8890 CPU mit 18 Kernen 2.105 GFLOPS. Workstation-Grafikkarten erreichen keine höhere Leistung als die Grafikkarten aus dem Desktop-Segment.

Die Strategie, Grafikhardware zur beschleunigten Durchführung allgemeiner Berechnungen zu nutzen, wurde von Owens et al. [112] bis zu einer Veröffentlichung von J.N. England aus dem Jahr 1978 zurückverfolgt, in der ein für die Berechnung von Rastercangrafiken entwickelter Prozessor zur Berechnung interaktiver Fittings einer kubischen B-Spline-Fläche verwendet wurde [49]. Erste GPGPU-Arbeiten auf Basis heutiger Desktop-Grafikprozessoren befassten sich laut Owens et al. mit der Roboterbahnplanung [90] (1990), der Auswertung künstlicher neuronaler Netze [13] (1998) und der Berechnung von Voronoi-Diagrammen [72, 106] (1995,1999). Diese frühen Ansätze waren stets darauf angewiesen, die in Hardware realisierten Algorithmen der Rendering-Pipeline zu reinterpretieren, um diese für die Lösung allgemeiner Probleme nutzbar zu machen. Mit der schrittweisen Einführung programmierbarer Pipeline-Schritte wurde der Zugang zu diesen Ressourcen vereinfacht und die Anzahl von Anwendungen, welche sich GPGPU zu nutze machen, stieg stark an. Eine weitere Vereinfachung ergab sich mit der Anerkennung und aktiven Förderung des GPGPU-Ansatzes durch die Grafikkartenhersteller und das Angebot von Programmierschnittstellen wie CUDA³

³<http://www.nvidia.de/cuda>

(2007, Nvidia), Stream⁴ (2007, AMD), OpenCL⁵ (2009, Khronos Group) oder DirectCompute⁶ (2009, Microsoft), welche auf die parallele Durchführung allgemeiner Berechnungen auf Grafikprozessoren zugeschnitten sind.

Trotz der Verfügbarkeit dieser Programmierschnittstellen nutzt die in diesem Kapitel vorgestellte Implementierung der Beschichtungssimulation die Möglichkeiten der klassischen OpenGL-Renderingpipeline in Kombination mit speziell zu diesem Zweck konzipierten Shaderprogrammen, um die Simulationsberechnungen effizient durchzuführen. Wie im weiteren Verlauf dieses Kapitels ersichtlich werden wird, ist diese Herangehensweise für die Simulation von Beschichtungsvorgängen aufgrund ähnlicher algorithmischer Strukturen vorteilhaft. Ein Vergleich mit dem CUDA-basierten Raytracingsystem Nvidia Optix Prime in Kapitel 6.2.2 bestätigt diesen Vorteil darüber hinaus auch für die Simulationsgeschwindigkeit.

5.2 Globaler Simulationsablauf

Das Ziel der Simulation besteht darin, verschiedene Zielgrößen auf der Oberfläche eines Bauteils sowie für die Positionen des Spritzbrenners entlang einer gegebenen Roboterbahn zu berechnen. Diese Zielgrößen wurden im Einzelnen bereits in Kapitel 2.3.1 beschrieben. Die simulierten Oberflächengrößen erlauben dem Anwender oder einem gekoppelten Bahnplanungssystem die Beurteilung, ob durch den Prozess und insbesondere durch die gewählte Roboterbahn die gewünschten Toleranzen auf dem gesamten Bauteil eingehalten werden können beziehungsweise ob und an welchen Stellen mit Abweichungen auftreten. Im Rahmen einer automatischen Bahnoptimierung dienen diese Daten als Grundlage für die iterative Verbesserung der Roboterbahn. Primäres Ziel ist hier in der Regel das Erreichen einer gleichmäßigen Schichtdickenverteilung. Zusätzlich bieten die bahnbezogenen Zielgrößen die Möglichkeit, einzelne Stellen oder Intervalle der Roboterbahn zu ermitteln, an denen suboptimale Eingriffsbedingungen, wie zum Beispiel zu flache Spritzwinkel oder auch eine zu geringe Spritzeffizienz, vorliegen und diese gezielt zu korrigieren.

In der entwickelten Simulation werden die oberflächenbezogenen Zielgrößen an diskreten Messpunkten auf der Oberfläche einer gegebenen Bauteilgeometrie berechnet, welche identisch mit den Vertices \mathbf{v}_j des zur Repräsentation der Bauteilgeometrie verwendeten Dreiecksnetzes sind. Die bahnbezogenen Zielgrößen werden für diskrete Zeitpunkte t_i in konstantem Abstand Δt der zeitlich parametrisierten robotergeführten Bewegungsbahn des Spritzbrenners berechnet. Die genaue Repräsentation der Bauteilgeometrie und der Roboterbahn wird im folgenden Kapitel 5.3 detaillierter beschrieben.

Basierend auf dieser Eingabe folgt der globale Simulationsablauf einem diskreten Zeitschritt-schema mit der Zeitschrittlänge Δt , in dem der Spritzbrenner entlang der gegebenen Roboterbahn bewegt und alle Zielgrößen inkrementell berechnet werden. Abbildung 5.1 verdeutlicht dieses Ablaufschema.

⁴<http://www.amd.com/de-de/innovations/software-technologies/stream>

⁵<http://khronos.org/opencl/>

⁶<http://www.microsoft.com>

Innerhalb eines Zeitschritts werden zunächst für den aktuellen Simulationszeitpunkt die Position und Ausrichtung des virtuellen Spritzbrenners anhand der gegebenen Roboterbahn ermittelt. Anschließend wird diejenige Teilmenge der Messpunkte der vernetzten Bauteiloberfläche bestimmt, für die in diesem Zeitschritt eine Zielgröße aktualisiert werden muss. Dies sind diejenigen Punkte, welche im virtuellen Spritzkegel des Brenners liegen und nicht durch andere Teile der Bauteilgeometrie verdeckt werden. Hierfür werden GPU-basierte Methoden zur Sichtbarkeitsberechnung verwendet.

Sind alle betroffenen Punkte ermittelt, werden die oberflächenbasierten Zielgrößen unter Inbegriffnahme auf die aktuelle Brennerposition aktualisiert. Insbesondere wird in diesem Schritt das gewählte Schichtauftragsmodell für jeden der Messpunkte ausgewertet und die in diesem Zeitschritt der Dauer Δt resultierende Schichtdicke wird zu der bereits akkumulierten Schichtdicke addiert. Je nach Wahl und Konfiguration eines der Schichtauftragsmodelle aus Kapitel 4 können verschiedene thermische Spritzprozesse mit unterschiedlichen Prozessparametrisierungen simuliert werden. Um eine einfache Anbindung einer thermodynamischen Simulation zur Berechnung von Bauteiltemperaturen zu ermöglichen, werden in diesem Schritt an den betroffenen Messpunkten auch oberflächenbezogene Randbedingungen ermittelt, welche die momentan an der Oberfläche anliegende Temperatur repräsentieren.

Im Folgenden kann optional ein Zeitschritt einer angekoppelten thermodynamischen Simulation ausgeführt werden, um die Bauteiltemperaturen zu aktualisieren. Das Ergebnis in Form aktualisierter Bauteiltemperaturen an den Netzknoten kann anschließend an die Beschichtungssimulation zurückgegeben und zur Berechnung weiterer, temperaturabhängiger Zielgrößen verwendet werden. Das Interface für die Integration einer thermodynamischen Simulation wird in Kapitel 5.6 erläutert. Eine entsprechende thermodynamische Simulationskomponente wurde in Kooperation mit dem Institut für Mechanik der TU Dortmund entwickelt und erfolgreich getestet [10], ist jedoch nicht Bestandteil dieser Arbeit.

Im letzten Berechnungsschritt der Beschichtungssimulation werden die bahnbezogenen Zielgrößen für den aktuellen Zeitschritt ermittelt. Da die bahnbezogenen Zielgrößen auf denselben Grundgrößen wie die oberflächenbezogenen Größen basieren, dienen in diesem Schritt die bereits ermittelten oberflächenbezogenen Daten sowie gegebenenfalls die Bauteiltemperaturen der letzten beiden Zeitschritte als Berechnungsgrundlage (vergleiche Kapitel 5.5.2).

Abschließend wird für den aktuellen Zeitschritt die Visualisierung der bisher berechneten Zielgrößen aktualisiert und die Simulationszeit wird um den gewählten Zeitschritt inkrementiert. Um die Effizienz der Simulation zu erhöhen, kann die Aktualisierungsfrequenz der Visualisierungen reduziert oder Visualisierungen während der Laufzeit der Simulation vollständig deaktiviert werden. Ebenso können die Berechnung der bahnbezogenen Größen sowie die thermodynamische Simulation deaktiviert werden, sofern die Ergebnisse dieser Schritte nicht benötigt werden. Die Berechnung der oberflächenbezogenen Größen ist jedoch Voraussetzung für die anderen Rechenschritte und kann nicht separat deaktiviert werden.

Bei Erreichen des Endes der Roboterbahn oder des eingestellten Simulationszeitintervalls wird die Simulation beendet. Die Ergebnisse können dann im Rahmen der Visualisierung, der Sensitivitätsanalyse oder der Parameteroptimierung betrachtet und weiterverwendet werden. Falls

die Simulation durch einen externen Bahnplanungsprozess oder eine andere Software automatisiert gesteuert wird, werden die Ergebnisse über die Netzwerkschnittstelle dem aufrufenden Prozess übermittelt. Nach der Auswertung der Ergebnisse kann dieser dann einen neuen Simulationsprozess, in der Regel mit veränderten Eingabedaten oder Simulationsparametern, initiieren.

5.3 Eingabedaten und Geometrirepräsentation

Wesentliche Eingaben für die Simulation sind die Geometrie des zu beschichtenden Bauteils, eine Roboterbahn, entlang welcher der Spritzbrenner bewegt wird, sowie eine den genauen Spritzprozess definierende Parametrisierung, deren Kern die Festlegung einer Massenflusscharakteristik für die Spritzpistole darstellt. Diese werden im Folgenden beschrieben.

5.3.1 Bauteilgeometrie

Die Geometrie eines Bauteils wird in der Regel mittels einer CAD-Software wie zum Beispiel Inventor⁷, CATIA⁸, Solidworks⁸ oder NX⁹ entworfen und liegt anschließend typischerweise als Begrenzungsflächenmodell (engl.: B-rep, Boundary representation) im Format der verwendeten CAD-Software vor. Für die Simulation ist, insbesondere vor dem Hintergrund der Verwendung GPU-basierter Beschleunigungstechniken, die Verwendung einer Dreiecksnetzrepräsentation sinnvoll, da Grafikprozessoren darauf spezialisiert sind, dreiecksbasierte Daten effizient zu verarbeiten.

Die Überführung der Bauteile in ein Dreiecksnetz erfolgt durch spezielle Triangulationsprogramme oder die CAD-Software selbst, welche in der Regel das STL-Format (Standard Tessellation Language) für den Export von Dreiecksnetzen unterstützen. Da an den Knoten des Dreiecksnetzes die Simulationszielgrößen berechnet werden, ist bei der Triangulierung darauf zu achten, dass die erzeugte Vernetzung der Oberfläche insbesondere auch in Bereichen niedriger Krümmung hinreichend fein ist und ausreichend viele Knoten aufweist. Viele CAD-Programme erzeugen jedoch Vernetzungen mit lokal krümmungsabhängiger Facettenanzahl unter Einhaltung eines maximalen Sehnenfehlers, so dass das resultierende Netz gegebenenfalls in einem weiteren Schritt adaptiv verfeinert werden muss (Abbildung 5.2). Hierfür genügt die Anwendung einfacher Facettenunterteilungsverfahren zur Verfeinerung planarer Bereiche des Netzes.

Das aus der Triangulation resultierende Netz

$$T = \left\{ \left\{ \mathbf{v}_j = (x_j, y_j, z_j)^T \right\}, \left\{ \mathbf{n}_j = ((n_x)_j, (n_y)_j, (n_z)_j)^T \right\}, \left\{ \mathbf{f}_k = (k_1, k_2, k_3) \right\} \right\}$$

wird in der Simulation als Knotenliste \mathbf{v}_j , Normalenliste \mathbf{n}_j sowie Dreiecks-Knoten-Inzidenzliste \mathbf{f}_k verwaltet, wobei die Indizes k_* die zu einer Dreiecksfacette inzidenten Vertex- und

⁷Autodesk, <http://www.autodesk.de>

⁸Dassault Systèmes, <http://www.3ds.com>

⁹Siemens AG, <http://www.siemens.com/nx>

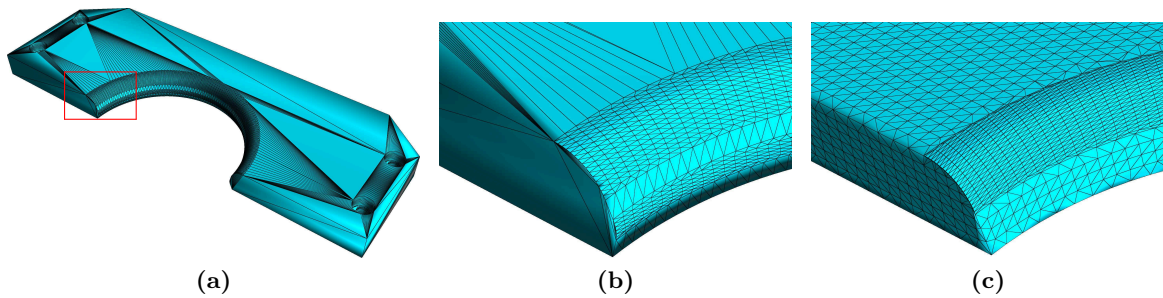


Abbildung 5.2: (a) Vernetzung eines Bauteils nach dem Exportieren aus einer CAD-Software (CATIA). (b) Vergrößerter Ausschnitt des Bauteils. Planare Teilflächen wurden hier mit möglichst wenigen Dreiecken vernetzt. Für die Simulation des Schichtauftrags liegen daher auf diesen Flächen kaum Simulationenpunkte vor. (c) Eine Vernetzung des Bauteils mittels eines speziellen Vernetzungsprogramms liefert auf allen Flächen eine gleichmäßigere Verteilung von Vertices.

Normalenindizes adressieren. Idealerweise werden die Normalenvektoren im Rahmen der Vernetzung eines CAD-Modells aus der CAD-Repräsentation berechnet und mit dem Dreiecksnetz gespeichert. Falls jedoch keine Normalenvektoren vorgegeben sind, werden diese für alle Vertices als arithmetisches Mittel der Normalen angrenzender Facetten berechnet, da diese für die Berechnung des Schichtauftrags benötigt werden.

Für die Verwendung einer thermodynamischen Simulation ist ein Oberflächennetz nicht ausreichend, da Temperaturen auch im Inneren des Bauteils berechnet werden müssen. Da in der Beschichtungssimulation die am Bauteil anliegenden Temperaturrandbedingungen an den Vertices des Oberflächennetzes berechnet werden, ist an dieser Stelle eine volumetrische, FEM-geeignete Triangulierung des Bauteils in Tetraederelemente, deren Randnetz dem verwendeten Oberflächennetz entspricht, gut geeignet. Unter einer FEM-geeigneten Triangulierung wird dabei eine Vernetzung verstanden, deren Elemente unter Berücksichtigung üblicher Qualitätsmaße für finite Elemente generiert werden [133]. Die Generierung einer entsprechenden Vernetzung kann beispielsweise durch die freie Software Tetgen¹⁰ auf Basis eines triangulierten Randnetzes erstellt werden [137], deren Ausgabeformat von der implementierten Beschichtungssimulation eingelesen werden kann. Auch für den Fall, dass die thermodynamische Simulation nicht durchgeführt werden soll, eignet sich dieser Ansatz zur Erzeugung eines gleichmäßigen Oberflächendreiecksnetzes.

Im Rahmen der Simulation werden die Bauteilnetze zur effizienten Visualisierung und Verarbeitung GPU-seitig in Form von Knotenkoordinaten- und Normalenliste sowie Dreiecks-Knoten-Inzidenzliste als Vertex Buffer Objects (VBO) gespeichert. Diese bieten insbesondere gegenüber klassischen Vertex Arrays den Vorteil, dass die Daten serverseitig gespeichert und verwaltet werden, so dass bei Zeichenbefehlen (`glDrawArrays*`, `glDrawElements*`) keine Daten aus dem Clientspeicher kopiert werden müssen, um sicherzustellen, dass die Daten aktuell sind. Zur weiteren Steigerung der Effizienz werden die Knoten-, Normalen- und Inzidenzlisten derart sortiert, dass Randknoten und Randelemente am Beginn der Vertex Buffer Objects

¹⁰<http://www.tetgen.org>

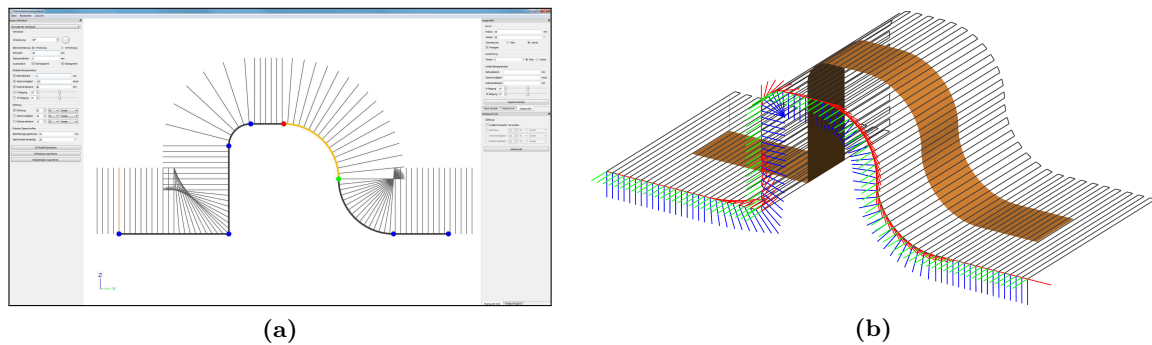


Abbildung 5.3: (a) Planung eines Bauteilquerschnitts und einer dazugehörigen Roboterbahn auf Basis von Geraden- und Kreissegmenten. (b) Resultierendes, extrudiertes Bauteil und mäanderförmige Roboterbahn [127].

stehen. Auf diese Weise können für oberflächenbezogene Berechnungen und Visualisierungen, trotz Verwendung eines Volumenmodells, identische Zeichenbefehle verwendet werden, bei denen lediglich die Anzahl der zu behandelnden Elemente beschränkt wird. Ineffiziente, nicht sequentielle Speicherzugriffe oder eine doppelte Datenhaltung werden so vermieden.

Für eine Bauteilgeometrie kann darüber hinaus ein beliebiger Teil ihrer Oberfläche als sogenannte Region of Interest (ROI) definiert werden, welche die geplante zu beschichtende Fläche beinhaltet. Verschiedene Module des entwickelten Simulationsframeworks, wie zum Beispiel die in Kapitel 5.8 beschriebene Parameteroptimierung und Sensitivitätsanalyse, verwenden diese ROI als Berechnungsgebiet für die Auswertung von Schichtfehlermaßen.

5.3.2 Roboterbahn

Die Roboterbahn des Spritzbrenners wird als zeitdiskrete Abfolge einer endlichen Anzahl an Roboterkonfigurationen beschrieben. Eine einzelne Konfiguration setzt sich aus der Position des TCP (engl.: Tool Center Point) im Werkstückkoordinatensystem (vergleiche Kapitel 3.4.1), der Ausrichtung des Brenners sowie dem Zeitpunkt, zu dem Position und Ausrichtung eingenommen werden, zusammen. Die Bahn kann daher als Folge von 8-Tupeln

$$\mathbf{p}_i = (x, y, z, q_1, q_2, q_3, q_4, t)_i \quad \text{bzw.} \quad \mathbf{p}_i = (\mathbf{x}_i, \mathbf{q}_i, t_i)$$

für $i = 1, \dots, n$ dargestellt werden, wobei \mathbf{x}_i die Position und \mathbf{q}_i die als Quaternion kodierte Ausrichtung bzw. Rotation des Brenners zum Zeitpunkt t_i bezeichnen. Im Rahmen der Simulation gilt stets $t_0 = 0$.

Während des Ablaufs der Beschichtungssimulation wird die Roboterbahn in diskreten Zeitschritten Δt startend bei $t = t_0$ abgetastet. Dabei kann Δt unabhängig vom zeitlichen Abstand $t_{i+1} - t_i$ aufeinanderfolgender Roboterkonfigurationen der Eingangsdaten gewählt werden. Für einen beliebigen Zeitpunkt t mit $t_i \leq t \leq t_{i+1}$ wird hierfür die Position des Spritzbrenners linear zwischen der Vorgänger- und der Nachfolgerkonfiguration interpoliert:

$$\mathbf{x}(t) = (1 - \tau)\mathbf{x}_i + \tau\mathbf{x}_{i+1}, \quad \tau = \frac{t - t_i}{(t_{i+1} - t_i)}. \quad (5.1)$$

Ebenso wird die Rotation – wie in der Roboterprogrammierung gebräuchlich – mit konstanter Winkelgeschwindigkeit und konstanter Rotationsachse (engl.: Spherical linear interpolation, Slerp) interpoliert [1, 32]. In Quaternionendarstellung ergibt sich dies nach [136] auf einfache Weise durch

$$\mathbf{q}(t) = \text{Slerp}(\mathbf{q}_i, \mathbf{q}_{i+1}, \tau) = \frac{\sin((1-\tau)\alpha)}{\sin\alpha} \mathbf{q}_i + \frac{\sin(\tau\alpha)}{\sin\alpha} \mathbf{q}_{i+1}, \quad (5.2)$$

wobei $\alpha = \cos^{-1}(\mathbf{q}_i \mathbf{q}_{i+1})$.

Roboterbahnen in dieser grundlegenden 8-Tupel-Darstellung können aus verschiedenen Quellen in die Simulation importiert werden. Realitätsnahe Bahnen können mittels der Roboterprogrammierungsumgebung ABB Robotstudio¹¹ bereitgestellt werden, deren Offline-Simulationsumgebung die Analyse und den Export von TCP-Bahnen und Orientierungen erlaubt. Die vergleichsweise hohe Genauigkeit dieser Bahnen resultiert dabei unter anderem aus der Berücksichtigung roboterspezifischer Dynamikberechnungen. Dabei werden an den Bahnpunkten definierte Zonendaten berücksichtigt, welche für jede Position eine Umkugelregion angeben, innerhalb derer der Roboter umorientiert und mit der Ansteuerung des nächsten Punktes beginnen kann. Je nach Eingriffssituation werden die Bahnpunkte dabei mit unterschiedlicher Genauigkeit passiert. Des Weiteren kann zur Erstellung von Roboterbahnen das im Rahmen einer Bachelorarbeit entstandene Bahnplanungstool verwendet werden [127]. Dieses erlaubt die simultane Erzeugung von extrudierten Bauteilgeometrien und der zugehörigen Bahnen basierend auf konfigurierbaren Querschnittsprofilen (Abbildung 5.3), welche sich aus gekoppelten Kreis- und Geradensegmenten zusammensetzen. Zusätzlich können Bahnen auch über die bereitgestellte TCP/IP-Schnittstelle der Simulation geladen werden.

5.3.3 Prozesskonfiguration

Da die entwickelte Beschichtungssimulation nicht auf einen bestimmten thermischen Spritzprozess oder auf eine festgelegte Parametrisierung eines Prozesses beschränkt ist, muss der zu simulierende reale Prozess durch Auswahl und Konfiguration eines Beschichtungsmodells definiert werden. Grundlage des entwickelten Footprint-basierten Beschichtungsmodells ist zunächst die Wahl einer der geometrischen Übertragungsfunktionen aus Kapitel 4.3 für die Simulation kegelförmiger oder zylindrischer Spritzstrahlcharakteristiken nach Gleichung (4.28) respektive (4.29). Im Rahmen des Simulationsframeworks entspricht dies der Wahl einer Funktionsdefinition in einem GLSL-Shaderprogramm, welche das gewünschte Modell implementiert.

Die eigentliche Festlegung einer Spritzstrahlcharakteristik geschieht durch die Wahl eines mittels der Methoden aus Kapitel 3 bestimmten Footprintprofils $\mathcal{F}(x, y)$ sowie der Stützstellen zur Modellierung des Haftwirkungsgrades $\xi_{\text{ang}}(\theta)$. Das Footprintprofil kann aus verschiedenen Scandatenformaten in die Simulation importiert, mit Zusatzinformationen versehen und in einem simulationseigenen Format abgespeichert werden. Zu den Zusatzinformationen gehört dabei neben der Spritzdauer $t_{\mathcal{F}}$ und dem Spritzabstand $d_{\mathcal{F}}$ auch die Festlegung des Ursprungs

¹¹ABB Ltd. (Asea Brown Boveri), <http://new.abb.com/products/robotics/robotstudio>

des Footprintkoordinatensystems, also der Position des Profils relativ zum TCP. Soll die thermodynamische Simulationskomponente zur Berechnung der Bauteiltemperaturen verwendet werden, so ist auch ein thermischer Footprint für die Berechnung der Randbedingungen erforderlich, welcher auf die gleiche Weise geladen oder als modifizierte bivariate Gaußverteilung erzeugt werden kann (vergleiche Kapitel 5.6). Der Abtastabstand der auf einem regulären Gitter in der x, y -Ebene angeordneten Stützstellen muss dabei identisch mit der von \mathcal{F} sein. Für die Simulation werden beide Footprintprofile in eine quadratische, GPU-seitig gespeicherte Floating-Point-Textur eingebettet, so dass der Ursprung des Footprintkoordinatensystems dem Mittelpunkt der Textur entspricht. Die von dieser Textur repräsentierte reale Kantenlänge (Höhe und Breite) im Werkstückkoordinatensystem wird mit $l_{\mathcal{F}}$ bezeichnet. Zugriffe auf die Footprinttextur mit Texturkoordinaten $(r, s) \in [0, 1]^2$ werden auf den gesamten Footprint abgebildet und liefern bilinear zwischen den Stützstellen interpolierte Profilwerte.

Die Stützstellen $(\theta_i, \xi_{\text{ang},i})$ der Haftwirkungsgradfunktion werden als tabellierte Daten importiert und mittels des in Kapitel 4.4.1 beschriebenen Hermite-Interpolationsverfahrens interpoliert. Eine hochauflösende Abtastung dieser Interpolierenden wird als eindimensionale Textur ebenfalls GPU-seitig gespeichert.

5.4 GPU-basiertes Simulationskonzept

Das Ziel des im Rahmen dieser Arbeit entwickelten GPU-basierten Simulationsansatzes ist die effiziente Berechnung der oberflächen- und bahnbezogenen Zielgrößen unter bestmöglicher Nutzung der parallelen Rechenleistung moderner Grafikkarten. Um die entwickelten Algorithmen und das komplexe Zusammenspiel der verschiedenen Shaderprogramme, welche diese Algorithmen realisieren, zu verstehen, ist es sinnvoll, zunächst die logische Struktur der Gesamtproblemstellung zu betrachten. Von dieser Betrachtungsweise können dann vereinheitlichte algorithmische Berechnungsschemata für die oberflächenbezogenen Zielgrößen einerseits und für die bahnbezogenen Zielgrößen andererseits abgeleitet werden. Diese Abstraktion von den konkret zu berechnenden Größen verdeutlicht überdies die einfache Erweiterbarkeit der Algorithmen auf potentielle weitere Zielgrößen.

Die Analyse der Problemstruktur ist Gegenstand des folgenden Unterkapitels. In den darauffolgenden Unterkapiteln wird anschließend das algorithmische Simulationskonzept zur GPU-basierten Berechnung der oberflächen- und bahnbezogenen Zielgrößen in schrittweise steigender Komplexität beschrieben, um die Motivation für die wichtigsten Entwicklungsschritte zu verdeutlichen und die Verständlichkeit der Beschreibung zu erhöhen.

5.4.1 Analyse der Problemstruktur

Tabelle 5.1 fasst zunächst die zu berechnenden und bereits in Kapitel 2.3.1 erläuterten Werte zusammen. Neben den eigentlichen geometrischen Zielgrößen – Schichtdicke, Spritzabstände und -winkel – gehören dazu auch die Eingangsrandbedingung für eine thermodynamische Simulation sowie die resultierenden Durchschnitts- und Extremtemperaturen in jedem Zeit-

	Oberflächenbezogene Werte	Bahnbezogene Werte
Zielgrößen	<ul style="list-style-type: none"> • Schichtdicke • max. Spritzwinkel • \emptyset gewichteter Spritzwinkel • min. Spritzabstand • \emptysetgewichteter Spritzabstand • max. Spritzabstand 	<ul style="list-style-type: none"> • Prozesseffizienz • Volumenablagerungsrate • max. Spritzwinkel • \emptyset gewichteter Spritzwinkel • min. Spritzabstand • \emptyset gewichteter Spritzabstand • max. Spritzabstand
Randbedingung Temperaturen	<ul style="list-style-type: none"> • Umgebungstemperatur 	<ul style="list-style-type: none"> • Max. Umgebungstemperatur • min. Temperatur • \emptyset Temperatur • max. Temperatur
Technische Hilfsgrößen	<ul style="list-style-type: none"> • Beschichtungszähler • Abgelagertes Schichtvolumen 	<ul style="list-style-type: none"> • Anzahl beschichteter Punkte

Tabelle 5.1: Übersicht über die mittels des entwickelten GPU-basierten Konzepts berechneten Werte, aufgeschlüsselt in Zielgrößen, Randbedingungen, temperaturbezogene Größen und Hilfsgrößen. Während die Temperaturrandbedingung sowie minimale, durchschnittliche und maximale Temperaturen im Rahmen dieses algorithmischen Konzepts berechnet werden, wird die Bauteiltemperatur selbst als Volumengröße in der optionalen thermodynamischen Simulation berechnet.

schritt. Zur Berechnung der Zielgrößen werden darüber hinaus weitere technische Hilfsgrößen mitgeführt, welche an gegebener Stelle beschrieben werden.

Da sowohl die oberflächen- als auch die bahnbezogenen Zielgrößen auf gemeinsame Grundgrößen – Schichtdicken- und Schichtvolumenzuwachs, Spritzwinkel und -abstand sowie die Bauteiltemperatur – zurückzuführen sind, ergibt sich für die Berechnung eine logische zweidimensionale Berechnungsstruktur die in Abbildung 5.4 dargestellt ist. Darin werden zunächst für jede Kombination aus Oberflächenmesspunkt \mathbf{v}_j und diskretem Simulationszeitpunkt t_i Grundwerte $f(\mathbf{v}_j, t_i)$ berechnet, wobei f an dieser Stelle als generischer Platzhalter für die fünf genannten Grundgrößen dient. So steht $f(\mathbf{v}_j, t_i)$ beispielsweise für den Spritzwinkel $\theta(\mathbf{v}_j, t_i)$ oder die Distanz $d(\mathbf{v}_i, t_j)$, unter dem der Oberflächenpunkt \mathbf{v}_j von der zum Zeitpunkt t_i des Prozesses aktuellen Brennerposition aus beschichtet wird. Dass hier stets mehrere zu berechnende Grundgrößen gemeint sind, ist in Abbildung 5.4 durch die Schichtung mehrerer Raster hintereinander verdeutlicht.

Auf Basis dieser Grundgrößen können sowohl oberflächen- als auch bahnbezogene Zielgrößen durch dieselben Operationen – Aufsummierung, Durchschnittsbildung oder Minimums- bzw. Maximumbestimmung – über alle Oberflächenpunkte respektive Zeitpunkte bestimmt werden. So ist beispielsweise die an einem Oberflächenpunkt erzielte Schichtdicke

$$H(\mathbf{v}_j) = H(\mathbf{v}_j, t_n) = \sum_{k=0}^n h(\mathbf{v}_j, t_k)$$

die Aufsummierung der Grundwerte h für den Schichtdickenzuwachs über die Zeit und der minimale Spritzabstand zu einem bestimmten Zeitpunkt t_i ist das Minimum der Spritzabstände aller Oberflächenpunkte \mathbf{v}_j mit positivem Schichtwachstum zu diesem Zeitpunkt:

$$D_{\min}(t_i) = \min_{\{k|h(\mathbf{v}_k, t_i)>0\}} d(\mathbf{v}_k, t_i).$$

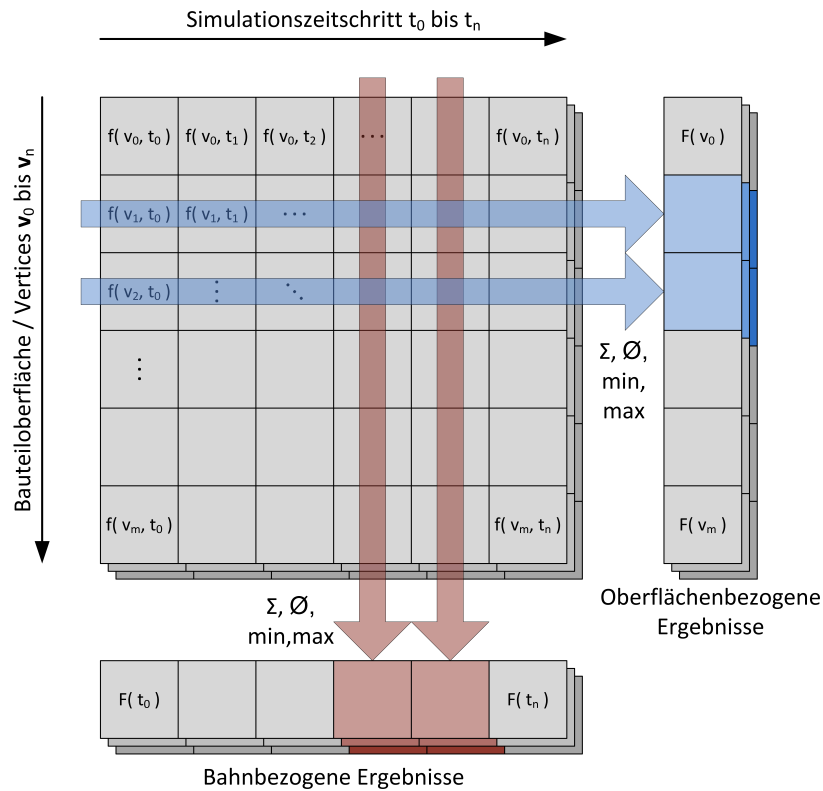


Abbildung 5.4: Logische, zweidimensionale Problemstruktur für die Berechnung oberflächen- und bahnbezogener Zielgrößen basierend auf den Grundwerten $f(\mathbf{v}_j, t_i)$ durch Summation, Durchschnittsbildung und Extremwertbildung.

Entscheidend für die Berechnung jedes einzelnen Grundwertes $f(\mathbf{v}_j, t_i)$ des Rasters in Abbildung 5.4 ist dabei, dass dieser nur dann für die Berechnung der Zielgrößen evaluiert werden darf, falls im betrachteten Zeitschritt t_i am betrachteten Vertex \mathbf{v}_j Material abgelagert wird, also $h(\mathbf{v}_j, t_i) > 0$ gilt¹². Dies ist genau dann der Fall, wenn der Oberflächenpunkt \mathbf{v}_j innerhalb des durch die aktuelle Brennerposition \mathbf{p}_i und den Simulationsfootprint \mathcal{F} definierten virtuellen Spritzkegels liegt, dem Brenner die durch den Vertex-Normalenvektor \mathbf{n}_j definierte Vorderseite zuwendet und nicht von Teilen der Bauteilgeometrie abgeschattet wird. Es handelt sich hier also um ein Sichtbarkeitsproblem, welches für jeden Vertex zu jedem Simulationszeitpunkt zu lösen ist. Formal kann dies im Rahmen der Berechnung des Schichtzuwachses durch Multiplikation mit einer Sichtbarkeitsfunktion $\chi(\mathbf{v}_j, t_i)$ ausgedrückt werden:

$$h(\mathbf{v}_j, t_i) = \chi(\mathbf{v}_j, t_i) \cdot h_{\mathcal{W}}.$$

Darin ist $h_{\mathcal{W}}$ das gewählte Beschichtungsmodell nach Gleichung (4.28) oder (4.29) und $\chi(\mathbf{v}_j, t_i) = 1$, falls \mathbf{v}_j von der Position des Spritzbrenners zum Zeitpunkt t_i sichtbar ist und $\chi(\mathbf{v}_j, t_i) = 0$ sonst.

¹²Zusätzlich wird für Minima und Maxima des Spritzabstandes und des Einschlagwinkels gefordert, dass \mathbf{v}_j zum Zeitpunkt t_i in dem in Kapitel 5.5.1 definierten Kernstrahlbereich liegt.

Bereits für eine mittelgroße Bauteilgeometrie mit ca. 50.000 Messpunkten und eine Roboterbahn mit 100.000 Zeitschritten muss die Sichtbarkeit von insgesamt 5 Milliarden Punkten berechnet werden. Für bisherige Simulationen auf diesem Gebiet stellte dies eine nicht effizient lösbare Aufgabe dar, so dass hier in der Regel starke geometrische Restriktionen für Roboterbahn und/oder Bauteil hingenommen wurden (vgl. Kapitel 2.3.2). Die Realisierung einer effizienten Beschichtungssimulation ohne derartige Restriktionen wird in dieser Arbeit im Wesentlichen durch die Ausnutzung der parallelen Rechenleistung moderner Grafikprozessoren für die Sichtbarkeitsberechnung ermöglicht.

5.4.2 Konzept zur GPU-beschleunigten Sichtbarkeitsberechnung

Die zentrale Idee besteht dabei darin, eine direkte geometrische Beziehung zwischen dem Spritzbrenner und dem Spritzkegel einerseits und der OpenGL-Kameraposition und der dazugehörigen Sichtpyramide andererseits herzustellen. Zur Veranschaulichung stelle man sich zunächst vor, durch die Düse des Spritzbrenners auf das zu beschichtende Bauteil zu blicken, wobei die Perspektive so gewählt ist, dass das Sichtfeld den gleichen Öffnungswinkel aufweist wie der Spritzkegel des Brenners, innerhalb dessen sich das Spritzmaterial bewegt. Unter dieser Bedingung sind die aus dieser Perspektive sichtbaren Teile des Bauteils diejenigen, die momentan beschichtet werden. Für die Simulation bedeutet dies, dass die sichtbaren Teile des Bauteils ermittelt werden können, indem zunächst das Zentrum der OpenGL-Kamera an den Koordinaten des virtuellen Spritzbrenners platziert, die Kamerablickrichtung entlang der Spritzachse ausgerichtet und der Öffnungswinkel sowie das Seitenverhältnis der Sichtpyramide so gewählt werden, dass es den virtuellen Spritzkegel des Brenners umfasst. Wird anschließend die virtuelle Bauteilgeometrie aus dieser Perspektive gerendert, werden nicht sichtbare Teile der Geometrie durch die hardwareseitig implementierten Backface-Culling-, Frustum-Culling- und Tiefenpuffer-Algorithmen entfernt. Durch eine Zuordnung der im Framebuffer verrasterten Vertices zur Bauteilgeometrie kann somit die Sichtbarkeit aller Vertices für einen Simulationszeitpunkt – entsprechend einer Spalte in Abbildung 5.4 – in einem einzelnen parallelisierten und hardwarebeschleunigten Berechnungsschritt bestimmt werden.

Um eine logische Zuordnung der im Framebuffer sichtbaren Teile des Bauteils zu Vertex-Indizes zu realisieren, liegt es nahe, Verfahren aus dem Bereich der Picking- bzw. Selection-Algorithmen zu verwenden. Ein geeignetes Verfahren ist das farbbasierte Picking, bei dem jedes Primitiv, oder in diesem Fall jeder Vertex, in einer eindeutigen RGBA-Farbe gezeichnet wird, die in den Index des Vertex umgerechnet werden kann. Insgesamt lässt sich daher die Berechnung der Sichtbarkeit und der Grundwerte für einen Zeitschritt t_i basierend auf der Grafikschnittstelle OpenGL mit folgendem, grundlegenden Algorithmus realisieren:

1. Bestimmung der Roboterposition $\mathbf{x}(t_i)$ und -ausrichtung $\mathbf{q}(t_i)$ gemäß der Gleichungen (5.1) und (5.2). Erstellen einer entsprechenden Model-View-Matrixtransformation zur Platzierung und Ausrichtung der Kamera gemäß $\mathbf{x}(t_i)$ und $\mathbf{q}(t_i)$.

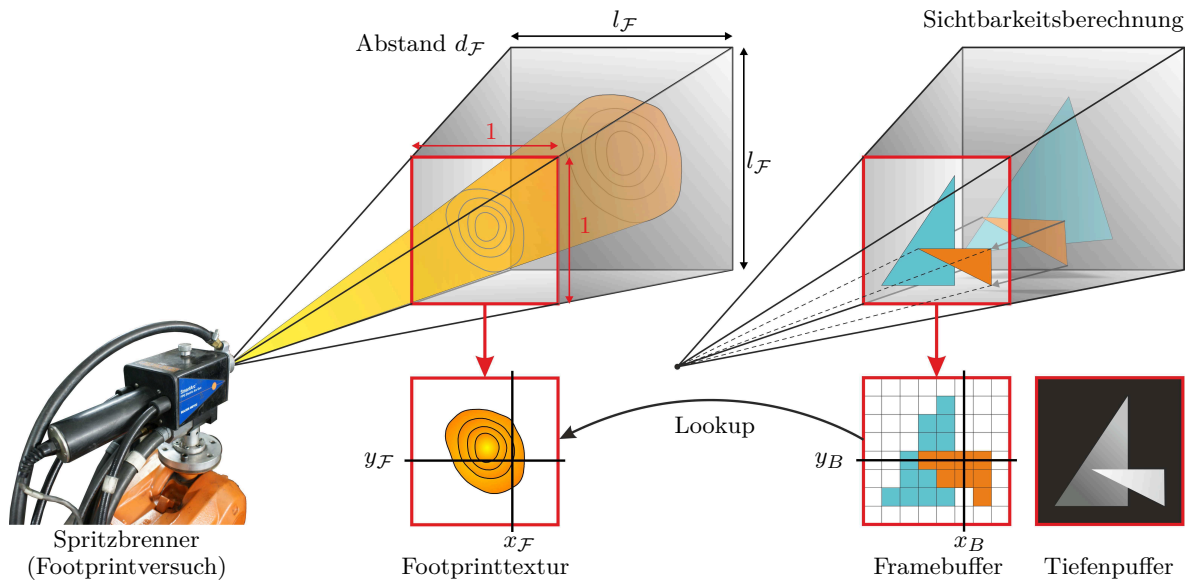


Abbildung 5.5: Dualität zwischen Spritzbrenner und Kamera beziehungsweise Spritzstrahl und Sichtpyramide. Durch die passende Auslegung der OpenGL-Sichtpyramide entsprechend der Maße des Footprints kann die Menge des in Richtung eines in der Bildebene verrasterten Messpunkts (x_B, y_B) abgestrahlten Materials anhand der Position (x_F, y_F) im Footprintprofil ermittelt werden.

2. Definition der quadratischen Sichtpyramide der Kamera durch Erstellen eines quadratischen Viewports (\rightarrow `glViewport`) und einer perspektivischen Projektionsmatrix mittels `gluPerspective`. Der vertikale und horizontale Öffnungswinkel der Sichtpyramide ergibt sich dabei mittels $2 \tan^{-1}(l_F / (2d_F))$ aus der Kantenlänge l_F und dem Spritzabstand d_F des verwendeten Footprints.
3. Zeichnen der im VBO abgelegten Bauteilgeometrie als ausgefüllte (\rightarrow `glPolygonMode(..., GL_FILL)`) Dreiecke mittels `glDrawElements(GL_TRIANGLES, ...)`, um den Tiefenpuffer vollständig zu initialisieren. Dies bedeutet, dass der Tiefenpuffer nach diesem Schritt bereits ein vollständiges Tiefenbild der unverdeckten Teile der Szene enthält. Der Framebuffer wird dabei mittels `glColorMask` schreibgeschützt und bleibt leer.
4. Erneutes Zeichnen der Bauteilgeometrie als einzelne Punkte in vordefinierten, eindeutigen Farben mittels `glDrawArrays(GL_POINTS, ...)`. Aufgrund des bereits vollständig initialisierten Tiefenpuffers werden alle Vertices, die nicht im Vordergrund sichtbar sind, automatisch entfernt. Da die Vertices Teil der im ersten Schritt zur Initialisierung des Tiefenpuffers gezeichneten Oberflächen sind, weisen die vordersten Vertices dieselben Tiefenwerte auf, die bereits an diesen Stellen im Tiefenpuffer stehen. Um sicherzustellen, dass diese Punkte nicht aufgrund von z-Fighting entfernt werden, wird beim Zeichnen des Tiefenpuffers im vorherigen Schritt ein negatives `glPolygonOffset` angewendet. Dadurch werden die Tiefenwerte der Oberfläche in Relation zu denen der Punkte minimal „nach hinten“, also von der Kamera bzw. dem Brenner weg, versetzt.

5. Auslesen des Framebuffers mittels `glReadPixels` und Ermitteln aller nicht hintergrundfarbenen Pixel und ihrer Koordinaten $(x_{\text{FB}}, y_{\text{FB}})$ im Framebuffer. Die zu diesen Farben gehörenden Vertices sind die sichtbaren Messpunkte.
6. Da der Framebuffer und die Sichtpyramide in OpenGL stets rechteckig, der Spritzstrahl aber näherungsweise kreisförmig im Querschnitt ist, muss in diesem Schritt für jeden gefundenen Messpunkt geprüft werden, ob tatsächlich Material in die exakte Richtung dieses Punktes vom Brenner abgestrahlt wird. Da die Sichtpyramide der Kamera anhand der Kantenlänge $l_{\mathcal{F}}$ und des Spritzabstandes $d_{\mathcal{F}}$ des Footprints derart ausgelegt wurde, dass der Footprint in einer Entfernung $d_{\mathcal{F}}$ exakt umschlossen wird, können die zweidimensionalen Koordinaten des Messpunktes in der Bildebene (x_B, y_B) mittels einfacher bilinearer Interpolation in die Koordinaten $(x_{\mathcal{F}}, y_{\mathcal{F}})$ der Footprinttextur umgerechnet werden, um die Stelle des Footprints zu ermitteln, welche in der gleichen Strahlrichtung wie der Messpunkt lag (Abbildung 5.5). Ist der Wert des Footprintprofils an dieser Stelle gleich Null, wird kein Material in die Richtung des Messpunktes abgestrahlt und es findet kein Schichtzuwachs statt.
7. Berechne für alle sichtbaren Messpunkte mit positivem Schichtzuwachs die Grundwerte $f(\mathbf{v}_j, t_i)$: Der Schichtdickenzuwachs wird entsprechend des gegebenen Beschichtungsmodells berechnet. Der Volumenzuwachs ergibt sich daraus durch Multiplikation mit der von dem Vertex repräsentierten Oberfläche, welche als ein Drittel der Summe der inzidenten Dreiecksflächen berechnet wird. Spritzwinkel und -abstand ergeben sich aus einfachen geometrischen Beziehungen nach Abbildung 4.2 und die Temperaturrandbedingung für \mathbf{v}_j kann der Textur des thermischen Footprints, an derselben Stelle $(x_{\mathcal{F}}, y_{\mathcal{F}})$ wie $h_{\mathcal{F}}$ aus der Footprinttextur, direkt entnommen werden.

Diese algorithmische Umsetzung in OpenGL zeigt, wie die Dualität zwischen Spritzbrenner und Kamera beziehungsweise Spritzstrahl und Sichtpyramide ausgenutzt werden kann, um die Sichtbarkeitsberechnung für beliebige Bauteilgeometrien und Roboterbahnen hardwarebeschleunigt durchzuführen. Diese grundlegende Vorgehensweise bietet jedoch noch viel Potential für eine Steigerung der Berechnungsgeschwindigkeit. Ferner kann es bei diesem Ansatz vorkommen, dass sichtbare Messpunkte nicht korrekt erkannt werden, falls mehrere Vertices an derselben Stelle verrastert werden. Im Folgenden werden daher aufeinander aufbauende Weiterentwicklungen verschiedener Aspekte dieses Algorithmus beschrieben, welche schließlich zu dem finalen in dieser Arbeit konzipierten Algorithmus führen.

5.4.3 Minimierung von Speichertransfers

Die Laufzeit dieses Ansatzes wird im Wesentlichen durch die benötigte Zeit für den Transfer des Framebuffers aus dem GPU- in den CPU-Speicher mittels `glReadPixels` und das anschließende Suchen der Messpunktpixel in dem transferierten Puffer bestimmt. Die maximal erreichbaren GPU-CPU-Transferraten stellen für viele GPU-Anwendungen einen Flaschenhals dar, so dass ein gutes Management dieser Transfers notwendig ist, um die zur Verfügung

stehende Rechenleistung von GPU und auch CPU nicht durch das Warten auf Speichertransfers zu verschwenden. Die theoretisch maximale Übertragungsgeschwindigkeit einer mit 16 PCI-Express 2.0 Lanes angebundenen Grafikkarte liegt in einer Richtung zwar bei knapp 8 GB/s, Messungen unter Verwendung von `glReadPixels` haben jedoch eine maximal erreichbare Transferrate von etwa 0,9–1,2 GB/s offenbart. Für einen mittelgroßen Framebuffer der Größe 1.024×1.024 Pixel ergibt sich für RGBA-Tupel mit 4 Byte pro Kanal ein Transfervolumen von 4 MB, so dass pro Sekunde, selbst unter Vernachlässigung aller weiteren Transfers und Operationen, nicht mehr als 225–300 Framebuffer pro Sekunde in den Clientspeicher heruntergeladen werden können. Für eine Roboterbahn mit 100.000 Positionen bedeutet dies eine reine Transferzeit von 6–8 Minuten zuzüglich der Zeit zur Suche der Messpunktpixel im Framebuffer und zur Berechnung der Grund- und Zielwerte. Sowohl die für den Transfer als auch für die Suche benötigte Zeit könnte durch eine Verringerung der Framebuffer-Auflösung erreicht werden, allerdings steigt dann die Wahrscheinlichkeit, dass zwei projizierte Messpunkte der Bauteilgeometrie in der Bildebene der Kamera so eng beieinander liegen, dass sie auf den gleichen Pixel im Framebuffer abgebildet werden. In diesem Fall können nicht beide Punkte als sichtbar erkannt werden und ein Punkt bleibt fälschlicherweise unbeschichtet. Besonders problematisch ist dies für Teilstücke der Bauteiloberfläche, die unter flachem Winkel betrachtet werden, so dass sehr viele Messpunkte dieser Fläche auf einen sehr kleinen Framebuffer-Bereich abgebildet werden. In diesem Fall wird ein besonders hoch auflösender Framebuffer benötigt, um alle Messpunkte zu erfassen.

Ausgehend von der Beobachtung, dass ein Großteil der Transferdauer dafür verwendet wird, den „leeren Zwischenraum“ zwischen relevanten Messpunktpixeln zu übertragen und diese dann in einem meist nur spärlich besetzten Puffer zu suchen, wurde im Rahmen dieser Arbeit ein Verfahren entwickelt, um die in die Bildebene projizierten Messpunkte stets ohne Zwischenraum im Framebuffer zu verrastern. Dabei ist jedem Vertex des Bauteils eine feste Position im Framebuffer zugeordnet, welche durch eine einfache Indextransformation von eindimensionalen Vertex-Indizes in zweidimensionale Framebufferpositionen berechnet werden kann. Auf diese Weise wird zum einen kein „Leerraum“ übertragen und die Suche nach den Pixeln, an denen ein Messpunkt verrastert ist, ist nicht mehr notwendig. Stattdessen kann die Sichtbarkeitsinformation für jeden Oberflächenmesspunkt direkt durch Auslesen des Farbwertes an einer durch den Index des Messpunktes vorgegebenen Position ermittelt werden.

Eine Schwierigkeit bei dieser angestrebten Verwendung des Framebuffers als generisches zweidimensionales Datenarray besteht darin, dass der Tiefenpuffertest basierend auf der Position der Fragmente im Framebuffer durchgeführt wird und diese automatisch von der GPU berechneten Fragmentpositionen nicht veränderbar sind. Die Kernidee zur Realisierung dieses verbesserten Verfahrens besteht daher darin, den Tiefenpuffertest in den Vertexshader vorzuverlagern und anschließend die Position (`gl_Position`) jedes Vertex exakt so zu verschieben, dass genau an der gewünschten Framebuffer-Position ein einziges Fragment generiert wird, sofern der Vertex sichtbar ist. Die verrasterten Vertices können auf diese Weise ohne Zwischenräume dicht gepackt im Framebuffer abgelegt werden. Da diese neuen Pixelpositionen der Messpunkte keinen Bezug mehr zu dem zuvor initialisierten Tiefenpuffer haben, ist die

manuelle Durchführung des Tiefentests erforderlich. Um diesen zu realisieren, wird der Tiefenpuffer nicht wie üblich im Standardtiefenpuffer, sondern in einer Tiefentextur, welche an einen benutzerdefinierten Offscreen-Framebuffer gebunden ist, erzeugt. Auf diese Textur kann aus dem Vertexshader mittels üblicher Texturfunktionen zugegriffen und der Tiefenwert manuell mit dem des aktuellen Vertex verglichen werden. Vertices, die den manuellen Tiefenpuffertest nicht bestehen, können bereits im Vertexshader entfernt werden, indem `gl_Position.w = 0` gesetzt und der Vertex damit außerhalb des Clip Space positioniert wird.

Mit dem so modifizierten Algorithmus wird die notwendige GPU-CPU-Transferrate deutlich reduziert. Entsprechend des zuvor angeführten Beispiels einer Geometrie mit 50.000 Vertices sind beispielsweise nur noch ca. 0,2 MB pro Zeitschritt, also 20-fach weniger als mit einem Framebuffer der Größe 1.024×1.024 , zu übertragen. Synthetische Messungen der erreichbaren Transferleistung haben gezeigt, dass bei kleineren Puffern der Overhead stark ansteigt und in der Praxis ca. 1.000–1.200 Framebuffer dieser Größe pro Sekunde übertragbar sind. Für diese theoretische Anzahl von 1.000–1.200 Zeitschritten pro Sekunde hat auch die Berechnung der Grundwerte $f(\mathbf{v}_j, t_i)$ selbst wieder einen signifikanten Anteil an der Rechenzeit.

5.4.4 Vollständig GPU-basierte Simulation

Um teure Speichertransfers vollständig zu vermeiden und gleichzeitig die hohe parallele Rechenleistung der GPU weitergehend für die Simulation zu nutzen, liegt es nahe, nicht nur die Sichtbarkeitsberechnung sondern auch alle weiteren Berechnungen zur Bestimmung der Grundwerte $f(\mathbf{v}_j, t_i)$ und Zielgrößen auf der GPU durchzuführen. Da die Speicherung aller Grundwerte aus dem in Abbildung 5.4 dargestellten Raster – in dem zuvor genannten Beispiel würden dafür pro Grundwert mehr als 18 GB benötigt – GPU-seitig nicht möglich ist, wurde ein neues Berechnungsschema entworfen, welches in Abbildung 5.6 dargestellt ist. Dabei werden die oberflächenbezogenen Zielgrößen inkrementell online berechnet (blaue Pfeile), also für den Zeitpunkt t_{i+1} ausschließlich basierend auf den Zielgrößen zum Zeitpunkt t_i sowie den aktuellen Grundwerten $f(\mathbf{v}_j, t_{i+1})$. Diese funktionale Verknüpfung ist in Abbildung 5.6 mit

$$F(\mathbf{v}_j, t_{i+1}) = F(\mathbf{v}_j, t_i) \oplus f(\mathbf{v}_j, t_{i+1}) \quad (5.3)$$

symbolisch durch den Metaoperator \oplus dargestellt, welcher, je nach Zielgröße, einfache oder gewichtete Summation oder auch Minimums- oder Maximumsbildung darstellen kann. Auf diese Weise genügt es, jeweils die Zielgrößen des aktuellen und vorherigen Zeitschritts im Speicher zu halten. Hierfür werden zwei Framebuffer Objects (A und B) mit jeweils mehreren 32-Bit-Floating-Point-Texturen verwendet, in deren Farbkanälen die oberflächenbezogenen Werte gespeichert werden. Die Framebuffer Objects werden in den Zeitschritten alternierend gelesen und beschrieben. Strukturell wird dabei der Sichtbarkeitstest für einen Vertex sowie die Berechnung der aktuellen Grundwerte im Vertexshader und die Online-Akkumulation der Zielgrößen im Fragmentshader realisiert. Die entsprechenden Shaderprogramme und Algorithmen werden detailliert im folgenden Unterkapitel 5.5.1 beschrieben.

Die Berechnung der bahnbbezogenen Zielgrößen ist nicht direkt innerhalb derselben Shaderpi-

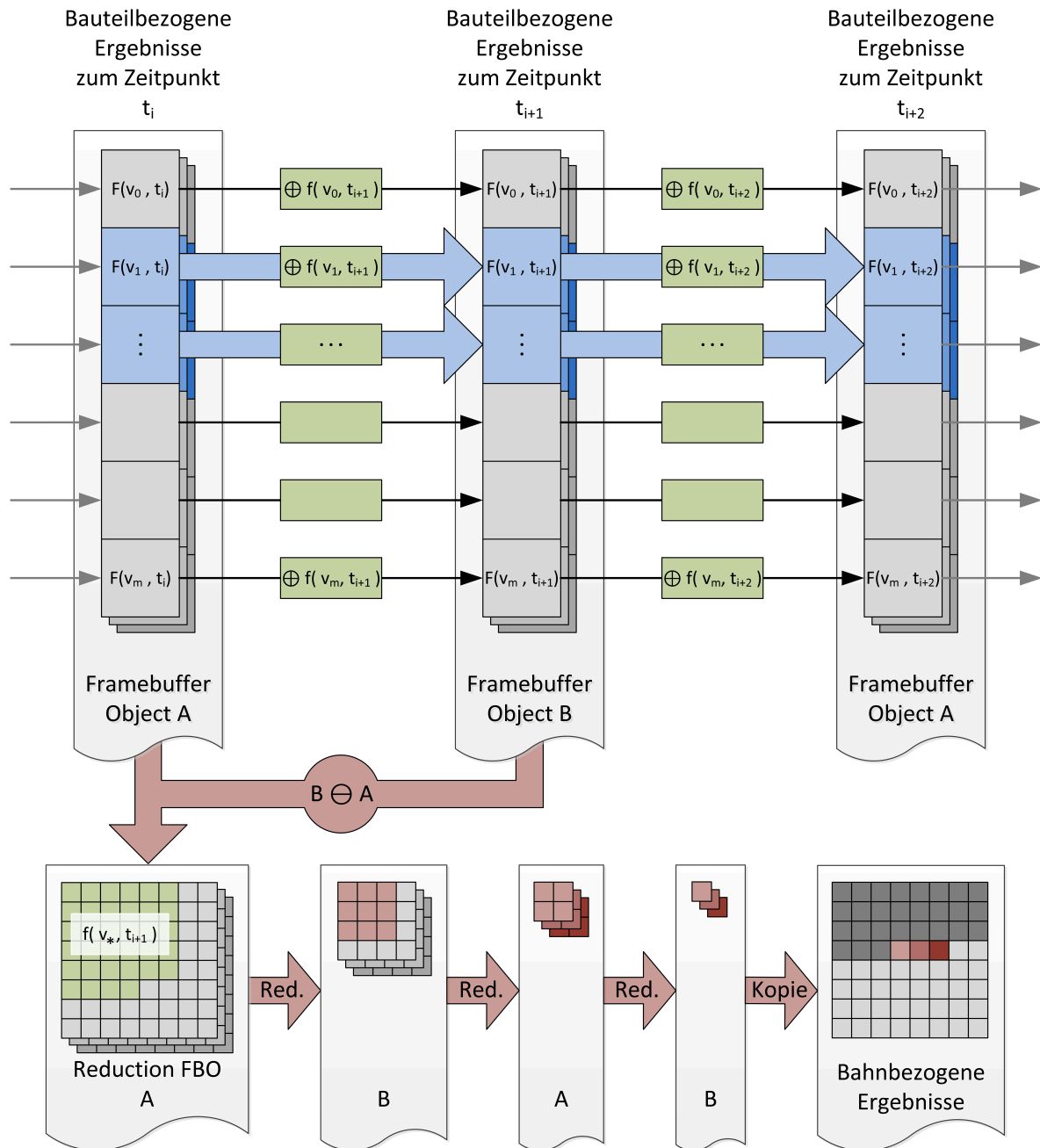


Abbildung 5.6: Übergeordneter Simulationsablauf und Datenfluss auf der GPU. Zur Speicherung der bis zum aktuellen Simulationszeitpunkt berechneten oberflächenbezogenen Zielgrößen kommen zwei Framebuffer Objects „A“ und „B“ zum Einsatz, welche alternierend beschrieben werden. Zur Berechnung der bahnbезogenen Zielgrößen (rote Pfeile) werden zunächst die momentanen Grundgrößen aus den letzten beiden Zeitschritten berechnet, mittels paralleler Reduktion auf je einen Wert reduziert und in einer separaten Textur gespeichert.

pipeline möglich, da die Shaderprogramme parallel für alle Vertices ausgeführt werden und in jedem Aufruf nur auf die Werte eines einzelnen Vertex zugegriffen werden kann. Die Berechnung der bahnbezogenen Zielgrößen erfolgt mittels einer separaten Shaderpipeline in einem zweiten Schritt. Dabei werden zunächst aus den beiden Framebuffers A und B, welche die Oberflächengrößen des aktuellen und des vorherigen Zeitschritts beinhalten, die momentanen Grundwerte $f(\mathbf{v}_j, t_i)$ zurückgerechnet und in verschiedenen Texturkanälen in einem gesonderten „Reduction FBO A“ zwischengespeichert. Als Resultat dieser Operation enthält ein Texturkanal den im aktuellen Zeitschritt addierten Schichtdickenzuwachs jedes Messpunktes, ein weiterer die Spritzwinkel aller Messpunkte usw.. Diese Werte werden anschließend mittels einer sogenannten parallelen Reduktionsoperation zu einem Wert reduziert. Dabei werden in jeder Iteration der Reduktion je vier Werte einer Textur als Minimum, Maximum, (gewichteter) Durchschnitt oder Summe zusammengefasst und in die Zieltextur geschrieben. Ein einzelner Reduktionsschritt halbiert dadurch die Texturgröße in Breite und Höhe. Dieses Verfahren wird solange durch Vertauschen von Quell- und Zieltexturen wiederholt, bis nur noch ein einziges Pixel pro Grundwert bzw. Texturkanal geschrieben wird, welches dann das Minimum, Maximum, den Durchschnitt oder die Summe der Grundwerte aller Oberflächenmesspunkte enthält. Die Ergebniswerte werden in eine Textur kopiert, in der die bahnbezogenen Zielgrößen für alle Zeitschritte gespeichert werden. Diese Shaderpipeline wird in Kapitel 5.5.2 genauer beschrieben.

5.5 Implementierung

Im Folgenden werden zentrale Teile der Implementierung, welche auf dem im vorherigen Abschnitt aufgestellten Simulationskonzept beruhen, vorgestellt. Der Schwerpunkt liegt dabei auf der Beschreibung der GPU-seitigen Implementierung der Shaderpipelines zur Berechnung der oberflächen- und bahnbezogenen Zielgrößen in den Kapiteln 5.5.1 und 5.5.2. Darüber hinaus wird in Kapitel 5.5.3 das Zusammenspiel zwischen CPU- und GPU-seitigem Code analysiert und es werden einige CPU-seitige Implementierungsdetails erläutert, welche zu einer signifikanten Steigerung der Simulationsgeschwindigkeit beitragen.

5.5.1 Berechnung oberflächenbezogener Zielgrößen

Die Shaderpipeline zur Berechnung der oberflächenbezogenen Zielgrößen besteht aus einem Vertexshader, in dem die momentanen Grundwerte für jeden Vertex berechnet werden, und einem Fragmentshader, in dem die Grundwerte mit den Zielgrößen des vorherigen Zeitschritts verknüpft werden. Abbildung 5.7 zeigt eine Übersicht über den Datenfluss in den Shadern, wobei insbesondere die Verteilung der Grund- und Zielgrößen auf die Farbkanäle der im Code verwendeten Variablen vom Typ `vec4` hervorgehoben ist¹³. So werden beispielsweise die Grundwerte Schichtzuwachs, Einschlagwinkel, Volumenzuwachs und Spritzabstand im Vertex-

¹³Aus Gründen der Übersichtlichkeit wurde auf einige Details sowie die Darstellung der Durchreichung der Temperaturrandbedingung in einer weiteren Textur verzichtet.

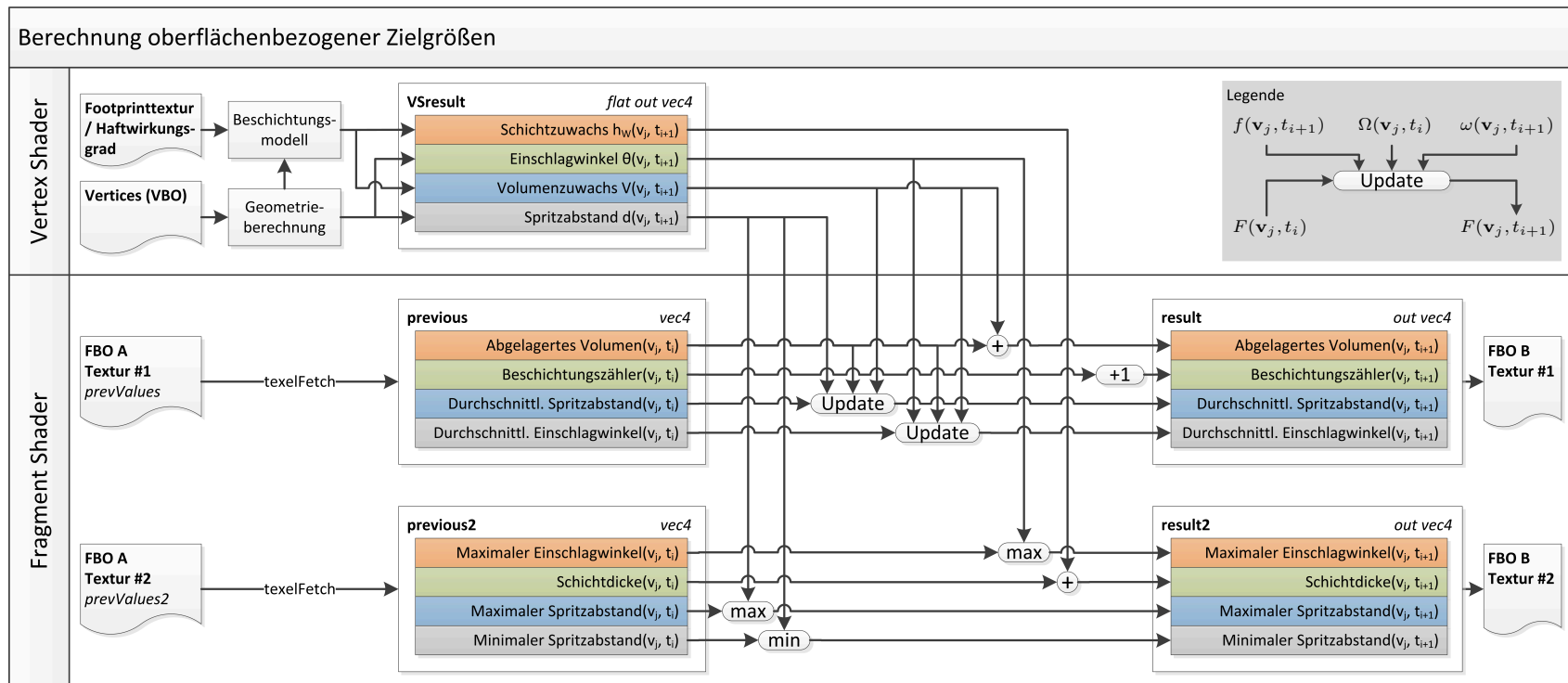


Abbildung 5.7: Shaderpipeline zur Berechnung oberflächenbezogener Zielgrößen. Im Vertexshader werden die Grundgrößen $f(v_j, t_i)$ berechnet und insbesondere das gewählte Beschichtungsmodell ausgewertet. Im Fragmentshader erfolgt die Akkumulation der Zielgrößen durch Verknüpfung mit den Werten aus dem vorherigen Zeitschritt.

shader in dieser Reihenfolge im Rot-, Grün-, Blau- und Alphakanal (in der Abbildung grau dargestellt) des Ausgabevektors `VSresult` („Vertex Shader result“) vom Typ `flat out vec4` abgelegt. Diese Werte werden auf Basis der Werte in der Footprinttextur, dem Haftwirkungsgrad und der Vertexposition berechnet.

Im Fragmentshader werden die Ergebnisse $F(\mathbf{v}_j, t_i)$ des vorherigen Zeitschritts aus den Texturen des Framebuffer Objects A geladen, mit den vom Vertexshader berechneten Werten verknüpft und anschließend als neue Zielgrößen $F(\mathbf{v}_j, t_{i+1})$ in den Zieltexturen des Framebuffer Objects B abgespeichert. Diese Verknüpfung „ \oplus “ für die inkrementelle Online-Berechnung der Zielgrößen ergibt sich für den maximalen Spritzwinkel sowie den minimalen und maximalen Spritzabstand durch

$$F(\mathbf{v}_j, t_{i+1}) = \{\max | \min\} (F(\mathbf{v}_j, t_i), f(\mathbf{v}_j, t_{i+t}))$$

und für die Schichtdicke und das abgelagerte Volumen durch einfache Addition

$$F(\mathbf{v}_j, t_{i+1}) = F(\mathbf{v}_j, t_i) + f(\mathbf{v}_j, t_{i+t}).$$

Die technische Hilfsgröße „Beschichtungszähler“ wird im Falle der Materialablagerung an dem Vertex um Eins inkrementiert und dient als Indikator, ob der Vertex beschichtet wurde oder nicht. Der durchschnittliche Spritzwinkel und -abstand werden mit dem abgelagerten Volumen gewichtet, um ein aussagekräftiges Ergebnis zu erhalten und die flächenmäßig großen Randbereiche des Footprints mit nur sehr geringem Materialauftrag nicht überzubewerten. Für die Verknüpfung „ \oplus “ wird an dieser Stelle daher eine Aktualisierungsvorschrift für gewichtete Mittelwerte benötigt. Der Mittelwert von $f(\mathbf{v}_j, t_i)$ gewichtet mit $\omega(\mathbf{v}_j, t_i)$ über alle Zeitschritte t_0, \dots, t_i am Knoten \mathbf{v}_j ist zunächst

$$F(\mathbf{v}_j, t_i) = \left[\sum_{k=0}^i \omega(\mathbf{v}_j, t_k) f(\mathbf{v}_j, t_k) \right] / \Omega(\mathbf{v}_j, t_i),$$

wobei

$$\Omega(\mathbf{v}_j, t_i) = \sum_{k=0}^i \omega(\mathbf{v}_j, t_k)$$

die Summe der Gewichte darstellt. Als inkrementelle Berechnungsvorschrift ergibt sich für den $(i+1)$ -ten Zeitschritt

$$\begin{aligned} F(\mathbf{v}_j, t_{i+1}) &= \left(\sum_{k=0}^{i+1} \omega(\mathbf{v}_j, t_k) f(\mathbf{v}_j, t_k) \right) / \Omega(\mathbf{v}_j, t_{i+1}) \\ &= \left(\omega(\mathbf{v}_j, t_{i+1}) f(\mathbf{v}_j, t_{i+1}) + \sum_{k=0}^i \omega(\mathbf{v}_j, t_k) f(\mathbf{v}_j, t_k) \right) / \Omega(\mathbf{v}_j, t_{i+1}) \\ &= \left(\omega(\mathbf{v}_j, t_{i+1}) f(\mathbf{v}_j, t_{i+1}) + \Omega(\mathbf{v}_j, t_i) F(\mathbf{v}_j, t_i) \right) / \Omega(\mathbf{v}_j, t_{i+1}) \\ &= \left(\omega(\mathbf{v}_j, t_{i+1}) f(\mathbf{v}_j, t_{i+1}) + (\Omega(\mathbf{v}_j, t_{i+1}) - \omega(\mathbf{v}_j, t_{i+1})) F(\mathbf{v}_j, t_i) \right) / \Omega(\mathbf{v}_j, t_{i+1}) \\ &= \frac{\omega(\mathbf{v}_j, t_{i+1})}{\Omega(\mathbf{v}_j, t_i) + \omega(\mathbf{v}_j, t_{i+1})} \cdot \left(f(\mathbf{v}_j, t_{i+1}) - F(\mathbf{v}_j, t_i) \right) + F(\mathbf{v}_j, t_i), \end{aligned} \quad (5.4)$$

welche nach dieser Umformung nur noch von den aktuellen Werten f und ω sowie den verfügbaren Größen F und Ω aus dem vorherigen Zeitschritt – gespeichert in `previous` und `previous2` in Abbildung 5.7 – abhängt. In Abbildung 5.7 entspricht die „Update“-Operation dieser Berechnungsvorschrift.

Vertexshader

In der Präambel des Vertexshaders werden nach der Definition der GLSL-Profilvariante und -version (4.20) zunächst technische Hilfsvariablen, Modellparameter, Textureinheiten und Ausgabevariablen definiert:

Vertex shader (Präambel)

```

1 #version 420 compatibility
2 #pragma optionNV(fastmath off)
3
4 // Technische Variablen
5 uniform int renderPass;           // 0: Tiefenpuffer, 1|2: Beschichten
6 uniform int bufSize;             // Dimension d. quadrat. Zielextextur
7 uniform float pixelIncrement;    // Pixelbreite in NDC; =2*(1/bufSize)
8
9 // Modellparameter
10 uniform float k_t;               // Zeitparameter dt/t_F
11 uniform vec3 footprintDimensions; // FP Breite/Höhe/neg. Spritzabstand
12 uniform float footprintFractionThreshold; // Grenzwert des Kern-Spritzstrahls
13
14 // Texturen
15 uniform sampler2D footprintLUT;  // Footprinttextur
16 uniform sampler1D depositionEfficiencyLUT; // Kennkurve Haftwirkungsgrad
17 uniform sampler2D depthBuffer;  // Vorberechneter Tiefenpuffer
18 uniform sampler2D prevValues2;  // Vorherige Ergebnistextur #2
19
20 // Ausgabe an Fragment Shader
21 flat out vec4 VSresult;          // Ergebniswerte für Textur #1
22 flat out vec4 VSresult2;        // Ergebniswerte für Textur #2

```

Die zentrale Steuerungsvariable `renderPass` gibt dabei an, ob in diesem Rendering-Durchlauf nur der Tiefenpuffer initialisiert wird (`renderPass=0`) oder ob dieser bereits initialisiert ist und die Sichtbarkeitsberechnung durchgeführt und die Zielgrößen berechnet werden können (`renderPass=1`). Der Spezialfall `renderPass=2` gibt an, dass kein Tiefenpuffertest durchgeführt werden soll und alle Vertices im Spritzstrahl als sichtbar behandelt werden. Dies beschleunigt die Berechnungen für Bauteile, die keine Selbstverdeckung aufweisen. Weitere Variablen definieren die Kantenlänge der quadratischen Zielextextur bzw. des Framebuffers (`bufSize`) sowie den Abstand benachbarter Pixel in normalisierten Gerätekoordinaten (`pixelIncrement`). Der Modellparameter `k_t` enthält den Quotient aus Zeitschritt und Footprintspritzdauer und der Vektor `footprintDimensions` in seinen drei Komponenten die halbe Breite und Höhe des Footprints sowie den bei der Footprinterzeugung verwendeten, negierten Spritzabstand a_z . Der Wert `footprintFractionThreshold` gibt einen Grenzwert zur Definition des Kerngebiets

des Spritzstrahls an, wobei Footprintprofilwerte $h_{\mathcal{F}}$ unterhalb dieses Grenzwertes sowie alle Raumpunkte entlang der zugehörigen Spritzrichtung als „außerhalb des Kernstrahls“ angesehen werden. Außerhalb des Kernstrahls werden Minima und Maxima des Spritzabstandes und des Einschlagwinkels nicht ausgewertet, da hier fast kein Spritzvolumen abgelagert wird. Ohne diese Beschränkung auf den Kernstrahl würden stark fluktuierende und schwer interpretierbare Werte resultieren.

Es folgt die Definition der Texturen, darunter die Footprinttextur, die eindimensionale Textur mit der Kennkurve des Haftwirkungsgrades gemäß Kapitel 4.4.1, gegebenenfalls der im vorherigen Rendering-Durchlauf vorberechneten Tiefenpuffer sowie eine der Texturen mit bisher berechneten Zielgrößen, von denen in diesem Shader zur Auswertung des Beschichtungsmodells lediglich die bis zu dem aktuellen Zeitschritt akkumulierte Schichtdicke benötigt wird. Die Ausgabe der berechneten Grundwerte erfolgt, wie in Abbildung 5.7 dargestellt, in den Vektoren `VSresult` und `VSresult2`. Da für die Ausgabe der Ergebnisse stets nur ein einziges Fragment verrastert wird, ist hier der Qualifizierer `flat out` ausreichend.

In der `main`-Methode werden zunächst die Clip-Space-Koordinaten `clipCoord` entsprechend der Standard-OpenGL-Transformationspipeline berechnet. Falls dieser Rendering-Durchlauf dem Initialisieren des Tiefenpuffers dient, werden die Koordinaten an den Fragmentshader weitergeleitet und der Vertexshader beendet, da die Szene in diesem Fall mit der normalen OpenGL-Pipeline gerendert wird (Zeilen 28–33):

Vertex shader, main-Methode, Teil 1/3

```

23 void main( void )
24 {
25     // Standard-Vertextransformation
26     vec4 clipCoord = ftransform();
27
28     // Pass 0: Weiterleiten an Fragmentshader und Tiefenpuffer
29     if (renderPass == 0)
30     {
31         gl_Position = clipCoord;
32         return;
33     }
34
35     // Berechne Vertexindex aus Farbkodierung
36     uint idx = uint( ( gl_Color.b * 255 * 65536 ) + ( gl_Color.g * 255 * 256 )
37                    + ( gl_Color.r * 255 ) + 0.5 );
38
39     // Verschiebe Vertex an neue Ziel-Fragmentposition
40     float x = -1.0 + ( ( float( idx % bufSize ) + 0.5 ) * pixelIncrement );
41     float y = -1.0 + ( ( floor( idx / bufSize ) + 0.5 ) * pixelIncrement );
42     gl_Position = vec4(x, y, 0.0, 1.0);
43
44     // Berechne Normalized Device Coordinates
45     vec3 NDC_pos = clipCoord.xyz/clipCoord.w;
46
47     // Berechne Texturkoordinaten und gl_FragDepth
48     vec3 textureAndDepthCoord = 0.5*NDC_pos + vec3(0.5,0.5,0.5);
49
50     // Footprinthöhe und -temperatur aus Textur ermitteln
51     vec2 footprint_lookup = texture2D(footprintLUT, textureAndDepthCoord.xy).rg;

```

```

51 float hf = footprint_lookup.r;
52 float temperatureBC = footprint_lookup.g;

```

Andernfalls wird zunächst der eindeutige eindimensionale Index `idx` des aktuellen Vertex aus den RGB-Farbkanälen der Vertexfarbe rekonstruiert (Zeile 36). Anschließend wird die zweidimensionale Zielposition der für diesen Vertex berechneten Größen im Framebuffer berechnet und die Ausgabevertexkoordinaten (`gl_Position`) an die entsprechende Stelle verschoben (Zeilen 38–41). Im nächsten Schritt werden die normalisierten Gerätekoordinaten `NDC_pos`, welche den Schnittpunkt des Strahls vom Kamerazentrum in Richtung des aktuellen Vertex mit der Bildebene der Kamera darstellen, ermittelt und in Texturkoordinaten für den Zugriff in die Footprinttextur umgerechnet. Vertices innerhalb der Sichtpyramide weisen dabei normalisierte Gerätekoordinaten innerhalb von $(-1,1)^2$ auf und werden auf den Texturkoordinatenbereich $(0,1)^2$ abgebildet. Gleichzeitig wird in der dritten Komponente des Vektors `textureAndDepthCoord` der Tiefenwert des resultierenden Fragments für den manuellen Tiefenpuffertest berechnet. Mit den Texturkoordinaten werden anschließend der Footprintwert `hf` und die Temperaturrandbedingung `temperatureBC` aus der Footprinttextur eingelesen. Der folgende Code-Abschnitt des Shaders zeigt die Sichtbarkeitsberechnung für den Vertex:

Vertex shader, main-Methode, Teil 2/3

```

53
54 // Berechnung abbrechen, falls Footprintwerte Null (Advanced Frustum Culling)
55 if ( ( hf == 0.0 ) && ( temperatureBC == 0.0 ) )
56 {
57     VSresult = vec4( 0 );
58     VSresult2 = vec4( 0 );
59     return;
60 }
61
62 // Tiefenpuffertest (falls angefordert)
63 if ( renderPass == 1 )
64 {
65     float referenceDepth = texture( depthBuffer, textureAndDepthCoord.xy ).r;
66     if ( referenceDepth < textureAndDepthCoord.z )
67     {
68         VSresult = vec4( 0 );
69         VSresult2 = vec4( 0 );
70         return;
71     }
72 }

```

Im ersten Schritt (Zeilen 54–60) werden dabei Vertices aussortiert, die außerhalb des Spritzstrahls liegen. Im Vorfeld der Simulation wurde hierfür die verwendete Textureinheit mittels

```

GLfloat color[4] = {0.0, 0.0, 0.0, 0.0};
glTexParameterfv(GL_TEXTURE_2D, GL_TEXTURE_BORDER_COLOR, color);
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_CLAMP_TO_BORDER);
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_CLAMP_TO_BORDER);

```

parametrisiert, wodurch Texturanfragen außerhalb des zulässigen Koordinatenbereichs auf den Wert Null abgebildet werden. Auf diese Weise wird das herkömmliche Frustum Culling, welches einen Vertex gegen die Seitenflächen der Sichtpyramide testet, mit der Abfrage der Footprinttextur kombiniert, um mit nur einer Abfrage automatisch alle Vertices – innerhalb oder außerhalb der Sichtpyramide – auszusortieren, in deren Richtung kein Material abgestrahlt wird bzw. für die keine Temperaturrendbedingung berechnet werden muss. Das herkömmliche Frustum Culling nach dem Vertex Shader ist zwar nicht abschaltbar und bleibt weiterhin aktiv, allerdings kann dadurch für die bereits in dieser Phase aussortierten Vertices die aufwendige Berechnung der Zielgrößen vermieden werden.

Im zweiten Schritt der Sichtbarkeitsberechnung (Zeilen 62–72) werden anschließend alle Vertices aussortiert, deren Tiefenwert größer ist als die Referenztiefe im vorberechneten Tiefenpuffer (`depthbuffer`) an der ursprünglichen Position `textureAndDepthCoord.xy` des Vertex. Für Vertices, die einen der Sichtbarkeitstests nicht bestehen, werden die Shaderausgabewerte auf Null gesetzt und der Shader beendet.

Eine interessante Alternative besteht an dieser Stelle darin, diejenigen Vertices, die einen der Sichtbarkeitstests nicht bestehen, durch Setzen von `gl_Position` auf eine Koordinate außerhalb des Clip Space, zu verschieben, so dass im Rahmen der Verrasterung kein Fragment generiert wird. Auf diese Weise wären deutlich weniger Aufrufe des Fragmentshaders erforderlich. Um in dieser Situation sicherzustellen, dass auch für diese Messpunkte die bisher berechneten Zielgrößen aus dem letzten Zeitschritt übernommen werden, können im Vorhinein alle Zielgrößenspeicher des vorherigen Schritts mittels `glCopyTexSubImage2D` in den Framebuffer kopiert werden. Insgesamt würden durch diese Änderung alle im Fragmentshader durchgeführten Kopieroperationen der Form $F(\mathbf{v}_j, t_{i+1}) \leftarrow F(\mathbf{v}_j, t_i)$ für nicht sichtbare Messpunkte durch eine einzelne, aber zeitaufwendigere Texturkopieroperation ersetzt. Laufzeitmessungen haben jedoch gezeigt, dass sich dieser Tausch in der Regel nicht lohnt. Insbesondere für kleinere Werkstückmodelle mit relativ wenigen Messpunkten ist der Overhead der Texturkopie deutlich zu hoch, aber auch für äußerst komplexe Werkstücke mit ca. 2 Millionen Messpunkten ist es geringfügig schneller, alle Kopieroperationen im Fragmentshader durchzuführen. Besteht der aktuelle Vertex die Sichtbarkeitstests und ist das abgestrahlte Material in dieser Richtung ($\rightarrow \mathbf{hf}$) nicht Null, kann nun das Beschichtungsmodell ausgewertet (Zeile 84) und als Ergebnis die Grundwerte zurückgegeben werden (Zeile 87):

Vertex shader, main-Methode, Teil 3/3

```
73 // Vertex sichtbar und Schichtdicke größer Null; Modell auswerten
74 if( hf != 0.0 )
75 {
76     float theta_imp;
77     float spray_distance;
78     float volume;
79
80     // Vorherige Schichtdicke aus Textur laden
81     float initial_hw = texture2D(prevValues2, vec2(0.5+0.5*x,0.5+0.5*y)).g;
82
83     // Beschichtungsmodell auswerten
```

```

84     float thickness = evaluateCoatingModel( NDC_pos.xy, hf, k_t, initial_hw ,
                                           theta_imp, spray_distance, volume );
85
86     // Ausgabe an Fragmentshader füllen
87     VSresult = vec4( thickness, theta_imp, volume, spray_distance);
88 } else {
89     // Schichtdicke ist Null
90     VSresult = vec4(0);
91 }
92
93 // Indikatorvariable zur Unterscheidung von Kern- und Randgebiet des Strahls
94 VSresult2.g = hf >= footprintFractionThreshold ? 1.0 : 0.0;
95
96 // Temperaturerandbedingung ausgeben
97 VSresult2.r = temperatureBC;
98 }

```

Die an `evaluateCoatingModel` übergebenen Variablen `theta_imp`, `spray_distance` und `volume` sind dabei Ausgabewerte der Auswertungsfunktion. Falls `hf` gleich Null ist, werden die Ausgabezielgrößen auf Null gesetzt (Zeile 90). Zuletzt werden eine Indikatorvariable, welche bei der Berechnung der bahnbezogenen Zielgrößen verwendet wird, um Kern- und Randgebiet des Spritzstrahls zu unterscheiden, sowie die thermische Randbedingung ausgegeben (Zeilen 94, 97).

Die Auswertung des kegelbasierten Beschichtungsmodells nach Gleichung (4.28) erfolgt in der Funktion `evaluateCoatingModel`, welche als Eingabe neben den normalisierten Gerätekoordinaten `NDC_pos`, dem Footprintwert `hf` und dem Zeitparameter `k_t` auch die bisher abgelagerte Schichtdicke `initial_hw` erhält:

Vertex shader (Beschichtungsmodell)

```

99 float evaluateCoatingModel( in vec2 NDC_pos, in float hf, in float k_t, in float
                             initial_hw, out float theta, out float sprayDistance, out float volume)
100 {
101     // Vertex entsprechend Schichtdicke in Normalenrichtung verschieben
102     vec4 surfacePtWrld = vec4(gl_Vertex.xyz + initial_hw * normalize(gl_Normal), 1);
103
104     // Transformation in Kamerakoordinaten
105     vec3 surfacePoint = (gl_ModelViewMatrix * surfacePtWrld).xyz;
106     vec3 n = normalize(gl_NormalMatrix * gl_Normal);
107
108     // Grundlegende Variablen
109     vec3 a = vec3(NDC_pos.xy, 1.0) * footprintDimensions;
110     float aDotN = dot(a, n);
111     float aLen = length(a);
112
113     // Einschlagwinkel
114     float theta_imp = acos( clamp(-aDotN/aLen, -1.0, 1.0) );
115
116     // Backface Culling
117     if(theta_imp > 1.570796326794)
118     {
119         theta = 0;
120         sprayDistance = 0;
121         volume = 0;

```

```

122     return 0.0;
123 }
124
125 // Haftwirkungsgrad aus Textur ermitteln
126 float xi = texture( depositionEfficiencyLUT, theta_imp * 0.636619772367 ).r;
127
128 // Distanzabhängiger Wirkungsgrad (optional)
129 // xi = xi * (1.184-0.001577 * length( surfacePoint ));
130
131 // Beschichtungsmodell auswerten
132 sprayDistance = length( surfacePoint );
133 double alpha = sprayDistance / length( a );
134 double radicand = ( pow( 1.0+hf/a.z, 3 ) - 1.0 ) * k_t * xi + alpha*alpha*alpha;
135 double hw = aDotN * ( newtonCubeRoot( radicand ) - alpha );
136
137 // Konvertierung zu einfacher Genauigkeit
138 float f_hw = abs(float(hw));
139
140 // Berechnung des abgelagerten Schichtvolumens
141 float aadn = float(alpha)*aDotN;
142 volume = ( length(gl_Normal) * ( pow(aadn+f_hw,3) - aadn*aadn*aadn ) )
           / ( 3 * (aadn - initial_hw)*(aadn - initial_hw) );
143
144 // Ausgabewert Einschlagwinkel
145 theta = degrees(theta_imp);
146
147 // Rückgabe der Schichtdicke
148 return f_hw
149 }

```

Mit Hilfe dieser Schichtdicke wird zunächst der Vertex in Richtung der Oberflächennormale verschoben, um die Verschiebung der Oberfläche aufgrund der bereits bestehenden Schichtdicke zu berücksichtigen (Zeile 102). Die resultierende Vertexposition sowie die Oberflächennormale `gl_Normal` werden dann in Kamerakoordinaten transformiert und zur Berechnung der für das Beschichtungsmodell benötigten geometrischen Terme verwendet (Zeilen 105–111, vergleiche Abbildung 4.2). Dazu gehört der Vektor `a` vom Kamera- bzw. Brennerzentrum zum Schnittpunkt mit der Footprintebene in Weltkoordinaten, die Länge `aLen` dieses Vektors, das Skalarprodukt $\langle \mathbf{a}, \mathbf{n} \rangle$ (`aDotN`) sowie der Einschlagwinkel `theta_imp`, bei dessen Berechnung in Zeile 114 das Argument der arccos-Funktion mittels `clamp` auf den gültigen Definitionsbereich beschränkt wird, um undefinierte Ergebnisse aufgrund von Fließkommaungenauigkeiten zu vermeiden. Der darauffolgende Codeabschnitt führt das insbesondere für die Beschichtung nicht geschlossener Geometrien wichtige Backface-Culling durch, indem für Vertices, deren Normale nicht dem Brenner zugewandt ist, für alle Rückgabewerte Null zurückgegeben wird. In Zeile 126 wird schließlich der vom Einschlagwinkel abhängige Haftwirkungsgrad aus der entsprechenden eindimensionalen Textur gelesen. Anschließend wird dieser optional mit dem distanzabhängigen Wirkungsgrad nach Gleichung (3.17) multipliziert. Da jedoch für den in dieser Arbeit verwendeten Lichtbogenspritzprozess keine signifikante Abhängigkeit von der Distanz nachgewiesen werden konnte (vergleiche Kapitel 4.4.3), ist diese Zeile an dieser Stelle auskommentiert. In den Zeilen 131–135 wird schließlich der Schichtwachstumswert `hw` entspre-

chend Gleichung (4.28) berechnet. Aufgrund der komplexen Verknüpfung von Variablen mit einem hohen Wertebereich hat es sich hier als vorteilhaft herausgestellt, diese Berechnungen mit doppelter Genauigkeit durchzuführen. Da GLSL jedoch keinen Befehl zur Berechnung von Kubikwurzeln doppelter Genauigkeit bereitstellt, wurde hierfür die Methode `newtonCubeRoot (double radicand)` implementiert, welche im Anhang B beschrieben wird.

Nach der Konvertierung des Ergebnisses `hw` in einen Wert einfacher Genauigkeit in Zeile 138 wird in den Zeilen 140–142 eine Näherung für das abgelagerte Schichtvolumen berechnet, welches später für die Berechnung der bahnbezogenen Prozesseffizienz und der volumengewichteten Zielgrößen benötigt wird. Ausgangspunkt für die Berechnung des Schichtvolumens $dV_{\mathcal{W}}$ ist Gleichung (4.15) für das werkstückseitige Kegelstumpfvolumen, wobei die Grundfläche $dA_{\mathcal{W}}$ des Kegels auf Basis der in der Länge `length(gl_Normal)` des Normalenvektors kodierten, von dem aktuellen Messpunkt repräsentierten Werkstückoberfläche dA_0 berechnet werden kann. Zu berücksichtigen ist hierbei, dass die beschichtete Oberfläche bereits der Spritzpistole entgegengewachsen ist, so dass die im Querschnitt des Spritzkegels beschichtete Fläche $dA_{\mathcal{W}}$ kleiner als dA_0 ist. Aus Abbildung 4.2 gehen die Abstände $-\alpha\langle\mathbf{a}, \mathbf{n}_{\mathcal{W}}\rangle$ und $-\alpha\langle\mathbf{a}, \mathbf{n}_{\mathcal{W}}\rangle + h_{\mathcal{W},\text{initial}}$ der beiden Ebenen, in denen $dA_{\mathcal{W}}$ respektive dA_0 liegen, von der Spitze des Spritzkegels hervor. Mit diesen ergibt sich nach Gleichung (A.3) aus Anhang A das Schnittflächenverhältnis

$$dA_{\mathcal{W}} = \left(\frac{-\alpha\langle\mathbf{a}, \mathbf{n}_{\mathcal{W}}\rangle}{-\alpha\langle\mathbf{a}, \mathbf{n}_{\mathcal{W}}\rangle + h_{\mathcal{W},\text{initial}}} \right)^2 dA_0.$$

Einsetzen in Gleichung (4.15) liefert damit den im Shaderprogramm implementierten Ausdruck

$$dV_{\mathcal{W}} = \frac{(\alpha\langle\mathbf{a}, \mathbf{n}_{\mathcal{W}}\rangle + h_{\mathcal{W}})^3 - \alpha^3\langle\mathbf{a}, \mathbf{n}_{\mathcal{W}}\rangle^3}{3(\alpha\langle\mathbf{a}, \mathbf{n}_{\mathcal{W}}\rangle - h_{\mathcal{W},\text{initial}})^2} dA_0$$

für das abgelagerte Volumen. Diese Volumenberechnung stellt für nicht planare Oberflächen nur eine Näherung dar, da das an einem Oberflächenpunkt abgelagerte makroskopische Volumen für lokal gekrümmte Oberflächen nicht Kegelstumpfförmig ist. Für eine hinreichend feine Vernetzung der Oberfläche ist diese Näherung für eine Abschätzung der Prozesseffizienz jedoch vollkommen ausreichend.

Abschließend wird in Zeile 145 des Shaderprogramms der Einschlagwinkel und anschließend das berechnete Schichtdickeninkrement zurückgegeben. Die Implementierung der Auswertungsfunktion für das einfachere zylindrische Beschichtungsmodell nach Gleichung (4.29) folgt demselben Schema; auf eine explizite Angabe der Funktion wird daher an dieser Stelle verzichtet.

Fragmentshader

Wie bereits in der Datenflussgrafik in Abbildung 5.7 dargestellt, werden die Ausgaben des Vertexshaders im Fragmentshader mit den Zielgrößen des vorherigen Zeitschritts verknüpft. Abgesehen von der bereits bekannten `renderPass`-Variable und den nun als Eingabe klassifizierten Ergebnisvektoren `VSresult` und `VSresult2` werden daher hier die Framebuffer-Texturen

`prevValues` und `prevValues2` mit den Zielgrößen aus dem vorherigen Zeitschritt sowie drei Ausgabevektoren `result`, `result2` und `result3` definiert:

Fragment shader (Präambel)

```

1 #version 420 compatibility
2 #pragma optionNV(fastmath off)
3
4 // Technische Variablen
5 uniform int renderPass;           // 0: Tiefenpuffer, 1|2: Beschichten
6
7 // Ergebnisse des Vertex Shaders
8 flat in vec4 VSresult;
9 flat in vec4 VSresult2;
10
11 // Texturen
12 uniform sampler2D prevValues;     // Vorherige Ergebnistextur #1
13 uniform sampler2D prevValues2;   // Vorherige Ergebnistextur #2
14
15 // Ausgabe an Framebuffertexturen
16 out vec4 result;
17 out vec4 result2;
18 out vec4 result3;

```

Die Ausgabevektoren `result` und `result2` sind dabei mittels `glBindFragDataLocation` an die Framebuffertexturen gebunden, die in der nächsten Iteration wiederum als Eingabe `prevValues` respektive `prevValues2` dienen. Da der dritte Ausgabevektor nur für einen Zeitschritt relevante Informationen speichert, wird hierfür keine dritte korrespondierende Eingabetextur benötigt.

Falls in diesem Rendering-Durchlauf nur der Tiefenpuffer initialisiert werden muss, wird in der `main`-Methode der Shader sofort beendet. Dadurch wird der von der Fixed-Function-Pipeline generierte Tiefenwert automatisch übernommen und es sind keine weiteren Berechnungen notwendig:

Fragment shader (main-Methode)

```

19 void main( void ) {
20     // Nur Initialisierung des Tiefenpuffers: keine Berechnung erforderlich
21     if (renderPass == 0) {
22         return;
23     }
24
25     // Bisher akkumulierte Ergebnisse auslesen
26     vec4 previous = texelFetch( prevValues, ivec2(gl_FragCoord.xy), 0 );
27     vec4 previous2 = texelFetch( prevValues2, ivec2(gl_FragCoord.xy), 0 );
28
29     // Kopiere zunächst Werte aus vorherigem Zeitschritt in die Ausgabe
30     result = previous;
31     result2 = previous2;
32     result3 = vec4( 0, VSresult2.r, 0, 0 ); // Temperaturrandbedingung
33
34     // Prüfe Fließkommaungenauigkeiten in der Volumenberechnung
35     precise float volume = previous.r + VSresult.b;
36     precise float machineEpsTest = volume - previous.r;

```

```

37  precise double err = abs(1.0 - (double(machineEpsTest) / double(VSresult.b)));
38
39  // Falls abgelagertes Volumen > 0 und FP-Genauigkeit ausreichend
40  if ( machineEpsTest > 0 && VSresult.b > 0 && err < 0.0001 )
41  {
42      result.r = volume; // abgelagertes Spritzvolumen
43      result.g = (previous.g + 1.0); // Zähler
44
45      // durchschnittlicher volumengewichteter Spritzabstand
46      result.b = ( VSresult.b * ( VSresult.a - previous.b )
47                  / ( previous.r + VSresult.b ) ) + previous.b;
48
49      // durchschnittlicher volumengewichteter Einschlagwinkel
50      result.a = ( VSresult.b * ( VSresult.g - previous.a )
51                  / ( previous.r + VSresult.b ) ) + previous.a;
52
53      // Akumulieren der Schichtdicke
54      result2.g = VSresult.r + previous2.g; // Schichtdicke
55  }
56
57  // Falls innerhalb des Kern-Spritzstrahls: berechne Min/Max-Größen
58  if ( VSresult2.g > 0 )
59  {
60      result2.r = max(VSresult.g, previous2.r); // maximaler Einschlagwinkel
61      result2.b = max(VSresult.a, previous2.b); // maximaler Spritzabstand
62      result2.a = min(VSresult.a, previous2.a); // minimaler Spritzabstand
63
64      result3.a = 1.0; // Kerngebiet des Strahls
65  }
66
67  // Thermo-Randbedingung in Ausgabe weiterleiten
68  result3.g = VSresult2.r;
69  }

```

Andernfalls werden zunächst mittels `texelFetch` die bisherigen Zielgrößen aus den Texturen geladen (Zeilen 26–27) und vorläufig, zusammen mit der Temperaturrandbedingung, in die Ausgabevektoren kopiert (Zeilen 30–32).

Falls der Schichtvolumenzuwachs in dieser Iteration größer als Null ist und der Messpunkt gegebenenfalls im Kernstrahlbereich liegt, werden im Folgenden die neuen Zielgrößen in den Komponenten der Vektoren `result` und `result2` entsprechend Abbildung 5.7 befüllt. Andernfalls werden die bisherigen Zielgrößen ohne Modifikation für den nächsten Zeitschritt übernommen. In jedem Fall wird die Temperaturrandbedingung aktualisiert (Zeile 66).

Bei der Akkumulation der Zielgrößen muss die auf 32 Bit begrenzte Fließkommagenauigkeit der Ausgabewerte gesondert beachtet werden, da es ansonsten vorkommen kann, dass der insbesondere in den Randbereichen des Spritzstrahls sehr kleine Schichtdicken- oder Volumenzuwachs bei der Akkumulation keine Änderung der Gesamtschichtdicke oder des Gesamtvolumens bewirkt. Dies passiert dann, wenn bei der Addition

$$F(\mathbf{v}_j, t_{i+1}) = F(\mathbf{v}_j, t_i) + f(\mathbf{v}_j, t_{i+1})$$

das Verhältnis $f(\mathbf{v}_j, t_{i+1})/F(\mathbf{v}_j, t_i)$ die relative Maschinengenauigkeit unterschreitet, welche für 32-Bit-Fließkommazahlen nach IEEE-754 Standard mit $\varepsilon_{\text{float}} = 2^{-23} \approx 1.19 \times 10^{-7}$ abge-

schätzt werden kann. In diesem Fall wäre dann $F(\mathbf{v}_j, t_{i+1}) = F(\mathbf{v}_j, t_i)$, obwohl $f(\mathbf{v}_j, t_{i+1}) > 0$ ist. Für die Akkumulation der oberflächenbezogenen Zielgrößen wäre der hierdurch gemachte Fehler zwar vernachlässigbar, bei der Berechnung der bahnbezogenen Größen ergeben sich dadurch jedoch schwerwiegende numerische Probleme. Diese sind darauf zurückzuführen, dass hierbei, gemäß des in Abbildung 5.6 dargestellten Berechnungsschemas, die Grundwerte $f(\mathbf{v}_j, t_{i+1})$ wiederum aus den Oberflächengrößen an aufeinanderfolgenden Zeitschritten ermittelt werden. Sind diese Grundwerte nicht oder nur mit hohem numerischem Fehler berechenbar, so ergeben sich auch hohe Abweichungen in den berechneten Bahngrößen und es können Divisionen durch Null auftreten, falls das Volumeninkrement zu Null berechnet wird. Kritisch ist insbesondere, dass bereits eine Division durch Null an einem einzigen aller Messpunkte ausreicht, um das gesamte Ergebnis unbrauchbar zu machen.

Da davon ausgegangen werden kann, dass in Strahlrandbereichen stets extrem kleine Werte auftreten können, würde eine Verdoppelung der Fließkommagenauigkeit, welche theoretisch unter signifikanter Reduktion der Performance erkaufte werden könnte, dieses Problem nicht lösen, sondern lediglich in den Bereich noch kleinerer Werte verschieben. Eine einfache Möglichkeit zur Lösung dieses Problems besteht darin, eine untere Schranke für das Volumeninkrement $f(\mathbf{v}_j, t_{i+1})$ festzulegen, unterhalb der $f(\mathbf{v}_j, t_{i+1})$ tatsächlich auf Null gesetzt und der Messpunkt für diesen Zeitschritt als „nicht beschichtet“ markiert wird. Als eine solche Schranke wäre $\varepsilon_{\text{float}} \cdot F_{\text{max}}$ geeignet, wobei F_{max} die maximal auftretende Schichtdicke ist. Der Nachteil dieser Methode besteht darin, dass der maximale Wert F_{max} für alle Vertices zuvor abgeschätzt werden muss und sich die minimal erlaubte Änderung damit stets an der höchstmöglichen Fehlerschranke orientiert, obwohl bei kleineren $F(\mathbf{v}_j, t_i)$ auch deutlich geringere Inkremente ohne numerische Probleme möglich sind. Dies führt insbesondere zu Beginn der Simulation sowie an dünn beschichteten Stellen zum unnötigen Verwerfen von Termen. Stattdessen wird in den Zeilen 35–37 des Shaderprogramms für jeden auftretenden Volumenzuwachs (`→VSresult.b`) geprüft, ob die Addition zum bisher abgelagerten Volumen `previous.r` und die anschließende erneute Subtraktion der Werte tatsächlich eine positive Änderung `machineEpsTest` des `float`-Wertes hervorruft. Ferner dient die in doppelter Genauigkeit berechnete Variable `err` der Abschätzung des Fehlers. Das verwendete spezielle Schlüsselwort `precise` stellt dabei sicher, dass die Berechnungen in exakt der angegebenen Reihenfolge durchgeführt werden und der GLSL-Compiler die Zeilen 35–37 nicht optimiert oder durch die logische Vereinfachung `machineEpsTest=VSresult.b` ersetzt. In Kombination mit der Compileranweisung `fastmath off` aus Zeile 2 wird sichergestellt, dass keine Compileroptimierungen zum Einsatz kommen, welche Abweichungen niederwertiger Bits für eine höhere Geschwindigkeit in Kauf nehmen. Die Bedingung in Zeile 40 stellt schließlich sicher, dass die für die Berechnung der bahnbezogenen Zielgrößen relevanten Werte nur dann inkrementiert werden, wenn das Volumeninkrement in der folgenden Berechnung der bahnbezogenen Größen wieder mit guter Genauigkeit rekonstruiert werden kann. Mit dieser Methodik wird die zur Verfügung stehende Fließkommagenauigkeit an jedem Oberflächenmesspunkt maximal ausgereizt. In den folgenden Zeilen 42–52 des Fragmentshaders werden Schichtvolumen und Schichtdicke akkumuliert und die gleitenden, volumengewichteten Mittelwerte von Spritzabstand und

Einschlagwinkel gemäß Gleichung (5.4) aktualisiert. Ferner wird die Hilfsvariable „Beschichtungszähler“ (\rightarrow result.g) um Eins inkrementiert. Für Vertices innerhalb des Kernbereichs des Spritzstrahls werden anschließend der maximale Einschlagwinkel sowie der minimale und maximale Spritzabstand aktualisiert und die Indikatorvariable result3.a auf Eins gesetzt. Auf Basis letzterer wird bei der anschließenden Berechnung der bahnbezogenen Zielgrößen zwischen Kern- und Randgebiet des Strahls unterschieden. Zuletzt wird die vom Vertexshader ermittelte Temperaturrandbedingung an die Ausgabe weitergeleitet.

5.5.2 Berechnung bahnbezogener Zielgrößen

Die Shader-basierte Berechnung der bahnbezogenen Zielgrößen erfolgt in den vier Schritten

1. Berechnung der Grundwerte $f(\mathbf{v}_j, t_{i+1})$ (Initialisierung),
2. Reduktion der Werte aller Messpunkte zu einer Summe oder einem Extremwert,
3. Normierung der Summen zur Bildung von gewichteten Mittelwerten und
4. Abspeichern der Ergebnisse.

Die Eingliederung dieser Schritte in den Simulationsablauf wurde bereits in Abbildung 5.6 verdeutlicht. Eine detailliertere Ablaufschema der Berechnung für die einzelnen Zielgrößen, an der sich die folgende Beschreibung orientiert, ist in Abbildung 5.8 dargestellt. Das zugehörige Shaderprogramm, welches die ersten drei Berechnungsschritte durchführt, ist der Vollständigkeit halber in Anhang C gegeben.

Initialisierung

Im ersten Schritt werden die benötigten Grundwerte $f(\mathbf{v}_j, t_{i+1})$ des aktuellen Zeitschritts aus den akkumulierten Oberflächengrößen $F(\mathbf{v}_j, t_{i+1})$ und $F(\mathbf{v}_j, t_i)$ des aktuellen und des vorherigen Zeitschritts berechnet. Diese Operation kann verallgemeinert durch den Metaoperator „ \ominus “ gekennzeichnet werden,

$$f(\mathbf{v}_j, t_{i+1}) = F(\mathbf{v}_j, t_{i+1}) \ominus F(\mathbf{v}_j, t_i),$$

welcher die Umkehrung der „ \oplus “-Operation für die Aufsummierung beziehungsweise die gewichtete Durchschnittsbildung darstellt. Für den Volumenzuwachs und den Beschichtungszähler $c(\mathbf{v}_j, t_{i+1}) \in \{0, 1\}$ ist dies die einfache Subtraktion

$$f(\mathbf{v}_j, t_{i+1}) = F(\mathbf{v}_j, t_{i+1}) - F(\mathbf{v}_j, t_i).$$

Der aktuelle Spritzabstand und Einschlagwinkel an einem Messpunkt werden aus den gespeicherten Oberflächenzielgrößen des durchschnittlichen, volumengewichteten Spritzabstands respektive Einschlagwinkels berechnet. Durch Umformen von Gleichung (5.4) ergibt sich hierfür:

$$f(\mathbf{v}_j, t_{i+1}) = \frac{\Omega(\mathbf{v}_j, t_{i+1})F(\mathbf{v}_j, t_{i+1}) - \Omega(\mathbf{v}_j, t_i)F(\mathbf{v}_j, t_i)}{\Omega(\mathbf{v}_j, t_{i+1}) - \Omega(\mathbf{v}_j, t_i)}.$$

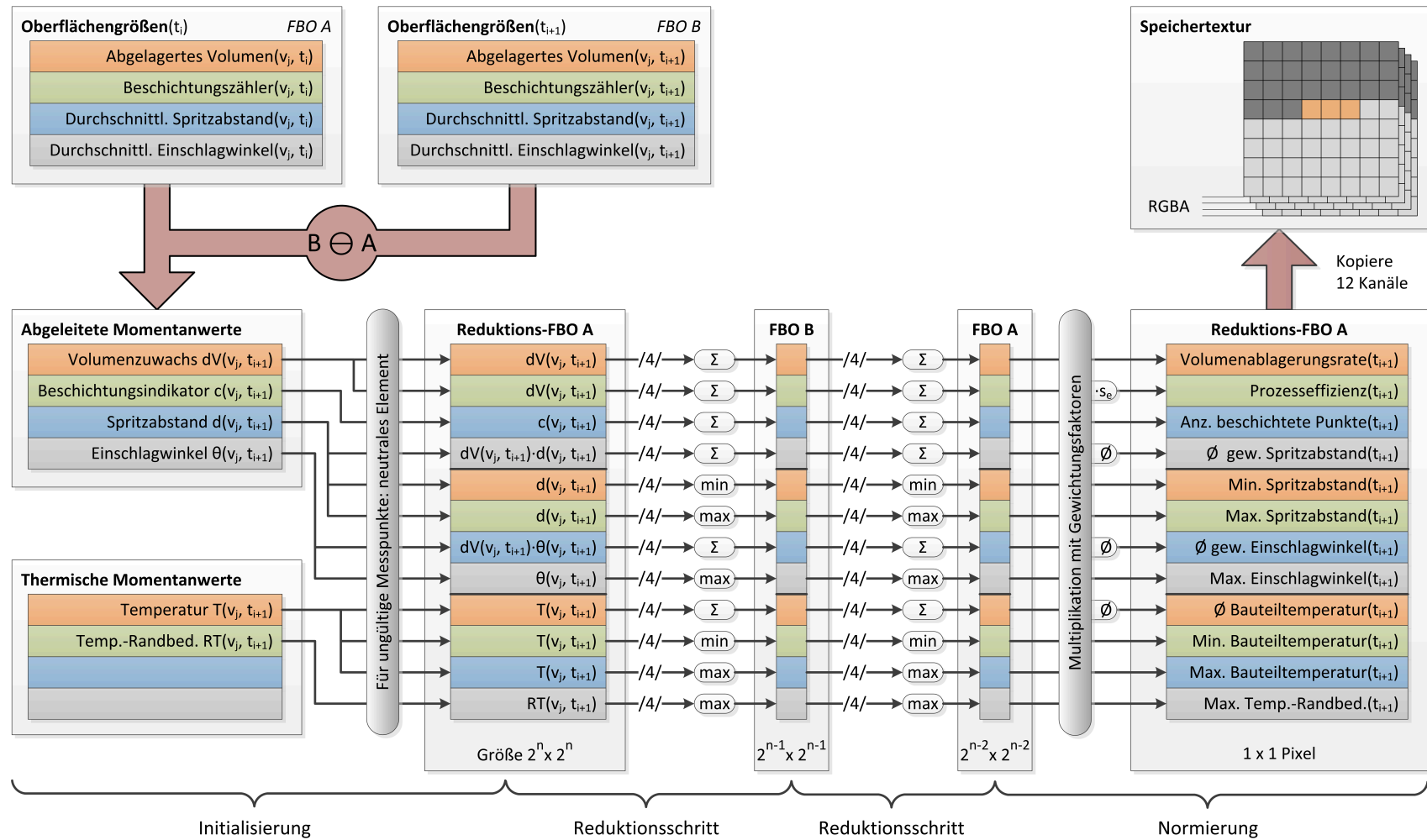


Abbildung 5.8: Shaderpipeline zur Berechnung Bahnbezogener Zielgrößen. In der Initialisierungsphase werden die momentanen Grundwerte aus den Oberflächengrößen des aktuellen und des vorherigen Zeitschritts berechnet, welche in den FBOs A und B gespeichert sind. Anschließend werden je Reduktionsschritt vier Werte zusammengefasst, wodurch sich Texturbreite und -höhe halbieren. Ist nur noch ein Wert übrig, wird dieser normiert und in die Speichertextur kopiert. Die gesamte Berechnung erfolgt gleichzeitig für zwölf Kanäle.

Zusätzlich zu diesen mittels „ \ominus “ abgeleiteten Grundwerten liegen Grundwerte für die Temperaturrandbedingung sowie gegebenenfalls die von der thermodynamischen Simulation berechneten Knotentemperaturen des aktuellen Zeitschritts direkt als Momentanwerte vor. Wie in Abbildung 5.8 dargestellt, werden mit diesen sechs Grundwerten die 3×4 Kanäle von drei RGBA-Texturen der Größe $2^n \times 2^n$ im Reduktions-FBO A durch das Shaderprogramm befüllt, wobei die Grundwerte entsprechend der Anzahl der aus ihnen berechneten Zielgrößen mehrfach Verwendung finden. Beispielsweise wird der Temperaturwert jedes Messpunktes in drei Kanäle kopiert, um im Folgenden in je einem Kanal die minimale, maximale und durchschnittliche Temperatur des Werkstücks zu berechnen. Da die Größe der Textur für die Reduktion eine Zweierpotenz sein muss, werden alle Texturwerte, die keinem Messpunktindex zugeordnet werden können, mit dem neutralen Wert der jeweiligen Reduktion initialisiert. Für die Bestimmung von Minima ist dies `FLOAT_MAX`, für Maxima `FLOAT_MIN` und für Summenreduktionen Null. Ebenso werden die Werte aller Messpunkte, welche nicht in die Berechnung einer Zielgröße einfließen sollen, auf den jeweiligen neutralen Initialwert gesetzt. Ausgeschlossen von der Reduktion werden dabei

- die inneren Volumenmesspunkte für alle Werte außer der Temperatur,
- die in diesem Zeitschritt unbeschichteten Messpunkte ($dV(\mathbf{v}_j, t_{i+1}) = 0$) für alle Werte außer der Temperatur und Randbedingung für eine thermodynamische Simulation,
- Messpunkte außerhalb des Kernstrahls für den minimalen und maximalen Spritzabstand, den maximalen Einschlagwinkel sowie die Anzahl beschichteter Punkte.

Letztgenannte Zielgrößen werden auf den Kernstrahlbereich beschränkt, um ein repräsentativeres Ergebnis zu erhalten und zu verhindern, dass Fluktuationen auftreten, falls weit außerhalb des Kernstrahls liegende Strahlbereiche das Bauteil streifen. Die Beschränkung der Auswertung der Anzahl beschichteter Punkte auf den Kernbereich des Spritzstrahls wiederum erlaubt im Nachhinein die Identifikation von Abschnitten der Roboterbahn, bei denen der Kernstrahl das Werkstück nicht trifft. Die übrigen Zielgrößen können damit leichter interpretiert werden.

Parallele Reduktion

Im zweiten Schritt wird die parallele Reduktion der zwölf Texturkanäle durchgeführt. Bei einer parallelen Reduktion werden zwei Reduktions-Framebuffer A und B alternierend beschrieben, wobei jeweils die Summe, das Minimum oder das Maximum von vier Texeln des Quell-Framebuffers in den Ziel-Framebuffer übernommen werden. Dabei halbieren sich jeweils die genutzte Breite und Höhe des Framebuffers, so dass nach nur $n - 1$ Schritten von einem initial $2^n \times 2^n$ Pixel großen Framebuffer nur noch ein Datum übrig bleibt, welches die Summe, das Minimum oder das Maximum aller ursprünglichen Werte beinhaltet. Durch die Unterteilung der Reduktion in viele Teiloperationen und deren parallele Ausführung auf den Shaderprozessoren der GPU ist dieses Verfahren sehr effizient. Für eine Umsetzung in

OpenGL wird in jedem der $k = 0, \dots, n - 1$ Schritte ein einzelnes Viereck gezeichnet, welches $2^{n-k+1} \times 2^{n-k+1}$ Pixel ausgehend von einer Ecke des Framebuffers überdeckt, so dass der Fragmentshader für jeden zu beschreibenden Zielpixel genau einmal aufgerufen wird. In diesem Shader wird die Reduktion, beispielsweise für eine Summe, mittels

```

...
ivec2 corner = ivec2(2.0*(gl_FragCoord.xy-0.5) + 0.5);
vec4 val1 = texelFetch(loopIn_Reduction1, corner, 0);
vec4 val2 = texelFetch(loopIn_Reduction1, corner + ivec2(1,0), 0);
vec4 val3 = texelFetch(loopIn_Reduction1, corner + ivec2(0,1), 0);
vec4 val4 = texelFetch(loopIn_Reduction1, corner + ivec2(1,1), 0);
Reduction1 = val1 + val2 + val3 + val4;
...

```

realisiert. Dabei ist `loopIn_Reduction1` eine Quelltextur des aktuellen Quell-Framebuffers und `Reduction1` ein Ausgabewert des Fragmentshaders vom Typ `vec4`. Die zu summierenden Quell-Textel `val1` bis `val4` werden hier aus einem 2×2 Texel großen Block mit dem Eckpunkt `corner` gelesen¹⁴.

Gewichtung

Im dritten Schritt werden die Ergebnisse der Reduktion mit Gewichtungsfaktoren multipliziert bzw. normiert. Dabei werden die gewichteten Summen des Spritzabstands und Einschlagwinkels durch das aufsummierte Volumen und die summierte Bauteiltemperatur durch die Anzahl der an der Summe beteiligten Messpunkte geteilt, um korrekte (gewichtete) Durchschnittswerte zu erhalten. Zusätzlich wird aus der Summe der abgelagerten Teilvolumina an den Vertices die Prozesseffizienz ermittelt, indem mit

$$s_e = \frac{t_{\mathcal{F}}}{\Delta t} \cdot \frac{1}{V_{\mathcal{F}}} \cdot 100\%$$

multipliziert wird, wobei $V_{\mathcal{F}}$ das Gesamtvolumen des Footprintprofils darstellt.

Speicherung

Für eine kompakte Speicherung der resultierenden bahnbezogenen Zielgrößen werden die zwölf Ergebniswerte schließlich im letzten Schritt aus dem Reduktions-FBO in einen Texturpuffer kopiert, welcher die in zeitlicher Abfolge abgelegten Bahnzielgrößen beinhaltet.

5.5.3 Optimierung der CPU-seitigen Implementierung

Durch die Implementierung der verschiedenen Berechnungsschritte der Simulation als Shaderprogramme wird nun fast der gesamte Rechenaufwand einer Simulation von der GPU geleistet. Neben der Optimierung des Shadercodes selbst sowie der vollständigen Vermeidung

¹⁴Fragment-Koordinaten liegen in OpenGL zwischen Ganzzahlen, also an Stellen 0,5, 1,5 etc.

von zeitintensiven Speichertransfers ist daher für die Minimierung der Simulationsdauer entscheidend, dass die Recheneinheiten der GPU möglichst gut ausgelastet werden und es keine Verarbeitungspausen gibt. Da die einzelnen Berechnungsschritte stets über das OpenGL-API von der CPU aus angestoßen werden, hängt die erreichbare GPU-Auslastung maßgeblich davon ab, wie schnell Befehle von der Simulationsapplikation über den Grafiktreiber und das Betriebssystem zur Grafikkarte transferiert werden. Schafft es die CPU, die CPU-seitigen Simulationsberechnungen und OpenGL-Aufrufe schneller zu verarbeiten und an die Befehlswarteschlange der GPU zu senden, als die GPU die Befehle abarbeiten kann, so ist die zu erwartende GPU-Auslastung hoch. Andernfalls muss die GPU zeitweise auf neue Befehle warten und die Simulationsdauer steigt.

Selbst bei einer schnellen CPU kann es jedoch zu Unterbrechungen in der Befehlsverarbeitungs-pipeline (Pipeline Stalls) kommen. Dies ist insbesondere dann der Fall, wenn seitens der Simulationsapplikation von der GPU berechnete Daten abgerufen oder Zustände der OpenGL-Zustandsmaschine mittels API-Befehlen abgefragt werden. Ein prominentes Beispiel für eine solche Datenabfrage ist der Abruf der berechneten Zielgrößen mittels `glReadPixels` zur Weiterverarbeitung und Auswertung auf der CPU. Dieser Befehl blockiert zunächst den Programmablauf auf der CPU solange, bis alle Befehle der Befehlspipeline, welche das Ergebnis noch beeinflussen können, abgearbeitet und die angefragten Ergebnisdaten in den CPU-seitigen Hauptspeicher transferiert worden sind. Läuft anschließend der Programmablauf auf der CPU weiter, müssen zunächst die Befehlsbuffer der Applikation wieder mit neuen OpenGL-Befehlen befüllt, diese vom Grafiktreiber übersetzt und an das Betriebssystem weitergereicht und schließlich von diesem in die Befehlswarteschlange auf der GPU transferiert werden. Insbesondere die Weiterleitung der Befehlspakete an die Hardwarequeue unterliegt dabei der alleinigen Kontrolle des Grafik-Schedulers des Betriebssystems und kann, je nach Auslastung des Systems, zu Verzögerungen führen. Während dieser gesamten Zeit wartet nun wiederum die GPU auf neue Befehle. Ein Pipeline-Stall kann jedoch auch in weniger offensichtlichen Situationen auftreten, wie zum Beispiel der Abfrage des Fehlerzustandes mittels `glGetError`, welcher voraussetzt, dass alle vorherigen Befehle abgearbeitet werden, bevor der aktuelle Fehlercode zurückgegeben werden kann.

Analyse des Simulationsablaufs

Um solche Situationen in der entwickelten Beschichtungssimulation zu vermeiden, wurde der Ablauf mit verschiedenen Werkzeugen, welche die OpenGL-API Aufrufe protokollieren, analysiert. Als Testwerkstück diente dabei eine Kugel mit 65.536 gleichmäßig auf der Oberfläche verteilten Messpunkten, welche von einer helixförmigen Bahn in 2.500 Zeitschritten beschichtet wird (Abbildung 5.9).

Abbildung 5.10 zeigt zunächst den zeitlichen Verlauf der Aufrufe der OpenGL-API eines einzelnen Zeitschritts, welcher mit der Nvidia Software Parallel Nsight¹⁵ aufgezeichnet wurde. Dabei wurden verschiedene, ineinander geschachtelte Zeitabschnitte der Simulation mittels

¹⁵<http://developer.nvidia.com/nvidia-nsight-visual-studio-edition>

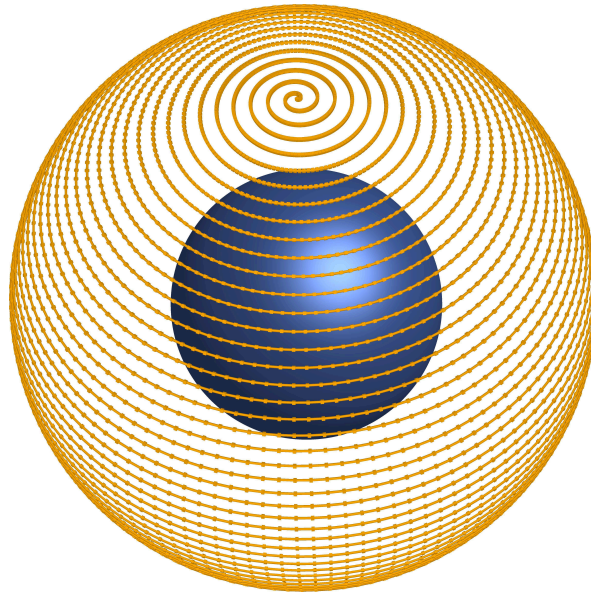


Abbildung 5.9: Helixförmige Beschichtungsbahn mit einem Abstand von 110 mm zur Oberfläche einer Kugel mit einem Durchmesser von 100 mm und 65.536 Messpunkten.

der Programmierbibliothek „Nvidia Tools Extension“ (NVTX) definiert, um die aufgezeichneten Aufrufe diesen Abschnitten zuordnen zu können. Innerhalb eines Zeitschritts erfolgen zunächst die vorbereitenden Schritte, in denen das Shaderprogramm und die FBOs gebunden sowie die Kamera entsprechend der aktuellen Spritzbrennerposition gesetzt werden. Die Berechnung der oberflächenbezogenen Zielgrößen erfolgt anschließend in den zwei Rendering-Durchläufen „Pass 0“ und „Pass 1“, welche in dieser Simulation zusammengenommen im Durchschnitt über alle Zeitschritte $30,4 \mu\text{s}$ dauern. Die darauffolgende Berechnung der bahnbezogenen Größen nimmt demgegenüber mit durchschnittlichen $83,5 \mu\text{s}$ deutlich mehr Zeit auf der CPU ein. In Abbildung 5.10 ist diese Berechnung weiter untergliedert in kurze Abschnitte zum Wechsel der Shaderprogramme und der Framebuffer zu Beginn und am Ende sowie die parallele Reduktion im Mittelteil. Letztere untergliedert sich wiederum in einen Initialisierungsschritt (vgl. Abbildung 5.8), zehn Reduktionsschritte und das Kopieren der Ergebnisse in die Speichertextur.

Auf einer mehrere Zeitschritte übergreifenden Ebene wurde des Weiteren eine Unterteilung in die Vorverarbeitung, die Hauptsimulationsschleife und die Nachverarbeitung vorgenommen (Abbildung 5.11, ③–⑥). Dabei beinhaltet die Vorverarbeitung das einmalige Laden und Kompilieren der Shaderprogramme sowie die Initialisierung der GPU-Datenstrukturen und die Nachverarbeitung beinhaltet das Herunterladen der Ergebnisse zur weiteren Verarbeitung sowie die Aktualisierung der während der Simulation deaktivierten Visualisierung. Um die reine Berechnungszeit der Simulation auf CPU und GPU ohne die Nachverarbeitungszeit messen zu können, wurde nach Beendigung der Hauptsimulationsschleife der Befehl `glFinish` eingefügt, welcher den weiteren Programmablauf verzögert, bis alle GPU-Befehle abgearbeitet worden sind.

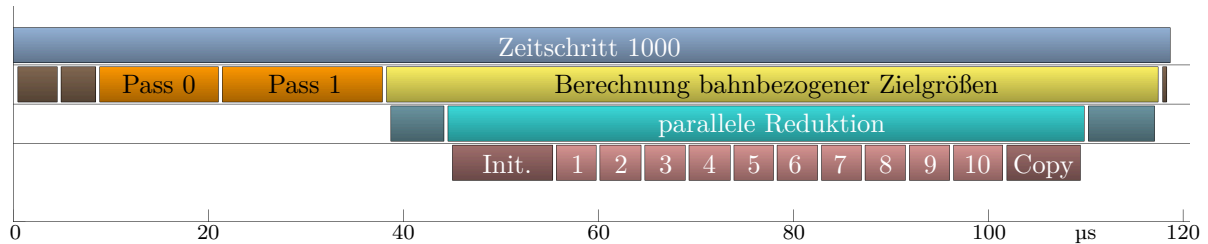


Abbildung 5.10: Zeitmessung der CPU-seitigen API-Aufrufe eines einzelnen Simulationsschritts (Nvidia Nsight). Innerhalb eines Zeitschritts entfällt der größere Teil der Zeit – in diesem Fall durchschnittliche $83,5\ \mu\text{s}$ – auf die Berechnung der bahnbezogenen Zielgrößen. Hierfür wurden in diesem Fall 10 Reduktionsiterationen durchgeführt. Die Berechnung der oberflächenbezogenen Zielgrößen in Pass 0 und Pass 1 dauert durchschnittlich $30,4\ \mu\text{s}$. Hinzu kommen die ungefähr konstanten Zeiten für die Initialisierung der Brennerposition und das Binden der Shaderprogramme und FBOs vor Pass 0 sowie vor und nach der parallelen Reduktion.

Abbildung 5.11 zeigt in den aufgezeichneten Simulationsläufen ③ und ④ vollständige Simulationen mit je 2.500 Zeitschritten, wobei in Verlauf ③ lediglich Oberflächenzielgrößen und in ④ auch Bahnzielgrößen berechnet wurden. Beide Verläufe basieren auf der finalen Simulationsversion, welche auch bereits das im nächsten Abschnitt vorgestellte Zustands-Caching enthält. Anhand dieser Simulationsläufe ist deutlich erkennbar, dass die Durchführung aller CPU-basierten Operationen inklusive aller Aufrufe des OpenGL-API in der Simulationsschleife bereits nach 106 bzw. 317 Millisekunden abgeschlossen ist, während die GPU für die Berechnung insgesamt 532 respektive 1.150 Millisekunden benötigt. Die Zeitdifferenzen von 426 bzw. 833 Millisekunden entsprechen genau den Zeiten, die die Simulation auf die Beendigung des Befehls `glFinish` wartet. Die Gesamtlaufzeit wird in diesem Fall also eindeutig durch die GPU-Leistung und nicht durch die CPU oder Speichertransfers beschränkt. Die Auslastung der GPU durch den Simulationsprozess beträgt mehr als 99%. Da die GPU in der Regel deutlich mehr Rechenleistung im Rahmen der Simulation erbringt, ist dieser Fall optimal. Zwar wird von der CPU in diesem Fall in ca. 70–80% der Zeit auf die GPU gewartet, in der keine Leistung erbracht wird, es ist jedoch kaum möglich diese freie Rechenkapazität gewinnbringend für die Simulation einzusetzen, da bereits der Transfer von wenigen Bytes zwischen GPU und CPU die freie Rechenzeit vollständig aufbrauchen würde.

Caching zur Eliminierung von Zustandsabfragen

Diese hohe GPU-Auslastung konnte, abgesehen von der Entfernung aller nicht unbedingt notwendigen Fehler- und Sicherheitsabfragen sowie redundanter Zustandswechsel der OpenGL-Zustandsmaschine, nur durch die Implementierung eines simulationsinternen Cachings wichtiger OpenGL-Zustände realisiert werden. Durch dieses Caching konnten alle unverzichtbaren Zustandsabfragen an den Grafiktreiber (\rightarrow `glGet*`) eliminiert und simulationsintern aufgelöst werden. Für die Ausführung der verschiedenen Simulations-Shaderprogramme werden dadurch vor allem die von OpenGL definierten Bindungspunkte für sämtliche Uniform-Variablen

(Uniform Locations) nur ein einziges mal nach jedem erforderlichen Linkvorgang des Shadercodes abgefragt und anschließend gecached, wodurch die Anzahl der API-Aufrufe erheblich reduziert werden konnte.

Um zu veranschaulichen, wie stark bereits ein einziger OpenGL-Befehl, welcher einen Teil des OpenGL-Zustands erfragt, die Leistung der Simulation beeinflussen kann und welche Bedeutung daher der Optimierung der einzelnen Befehle und dem Caching des OpenGL-Zustands zukommt, zeigt Abbildung 5.11 die zeitlichen Abläufe aufeinanderfolgender Zeitschritte. Während ① einige Zeitschritte des Verlaufs der mittels Zustands-Caching optimierten Simulation darstellt, zeigt ② zwei Zeitschritte eines Simulationslaufs, bei dem ein einzelner zusätzlicher Befehl `glGetUniformLocation` am Ende jedes Zeitschritts eingefügt wurde, welcher den Bindungspunkt einer Uniform-Variablen eines Shaderprogramms erfragt, um auf diese Variable zugreifen zu können. Dieser einzelne Befehl benötigt für die Ausführung mit $447\ \mu\text{s}$ in etwa genauso lange wie 3–4 vollständige Zeitschritte mit jeweils 172 API-Befehlen in Fall ①.

Den Einfluss dieses einzelnen `glGetUniformLocation`-Befehls auf den Ablauf der gesamten Simulation zeigt der aufgezeichnete Simulationsverlauf ⑤. Im Gegensatz zu dem optimierten Simulationsablauf ④ löst die Zustandsabfrage Verzögerungen bei der Verarbeitung der API-Befehle aus, da der Grafiktreiber zur Bestimmung des angeforderten Uniform-Bindungspunktes zunächst alle vorherigen Befehle betrachten muss. Ob dabei auf die GPU-seitige Ausführung der Befehle gewartet wird oder die Verzögerungen durch einen gestiegenen Berechnungsaufwand im Grafiktreiber verursacht werden, lässt sich ohne interne Kenntnisse des verwendeten Treibers nicht beantworten. Die Ausführung weiterer `glGetUniformLocation`-Aufrufe an derselben Stelle verzögert den Gesamtverlauf zwar weiter, aber nicht mehr so deutlich wie der erste Aufruf. Die nun vernachlässigbar geringe Dauer des `glFinish`-Aufrufs zeigt deutlich, dass die Simulationszeit in diesem Fall von der langsameren Verarbeitungsgeschwindigkeit der CPU beziehungsweise einem Pipeline-Stall diktiert wird.

Einfluss der Visualisierung auf die Laufzeit

Während einer Simulationsberechnung kann der Fortschritt der Beschichtung auf dem Bauteil sowie der zeitliche Verlauf der bahnbezogenen Zielgrößen visualisiert werden (vergleiche Kapitel 5.7). Obwohl diese Visualisierungen direkt die GPU-seitigen Datenstrukturen für die Zielgrößen benutzen und daher keine Speichertransfers notwendig sind, werden für das Neuzeichnen des Bauteils ca. 60 ms benötigt. Diese im Vergleich mit der Dauer von $125\ \mu\text{s}$ eines Simulationszeitschritts lange Zeit ist darauf zurückzuführen, dass für die Aktualisierung der Visualisierung eine große Anzahl an OpenGL-Zuständen modifiziert werden muss. Das Ablaufdiagramm ⑥ in Abbildung 5.11 zeigt den Ablauf der optimierten Simulation, wobei die Visualisierung in jedem 50. Zeitschritt aktualisiert wurde, was zirka 15 Frames pro Sekunde entspricht. Etwa 89 % der Rechenzeit entfallen hier auf die Visualisierung. Für Laufzeitmessungen und während der Steuerung der Simulation über die Netzwerkschnittstelle im Rahmen einer automatisierten Bahnplanung oder -optimierung werden Aktualisierungen daher vollständig deaktiviert. Im Rahmen einer manuell durchgeführten Bahnplanung oder -analyse

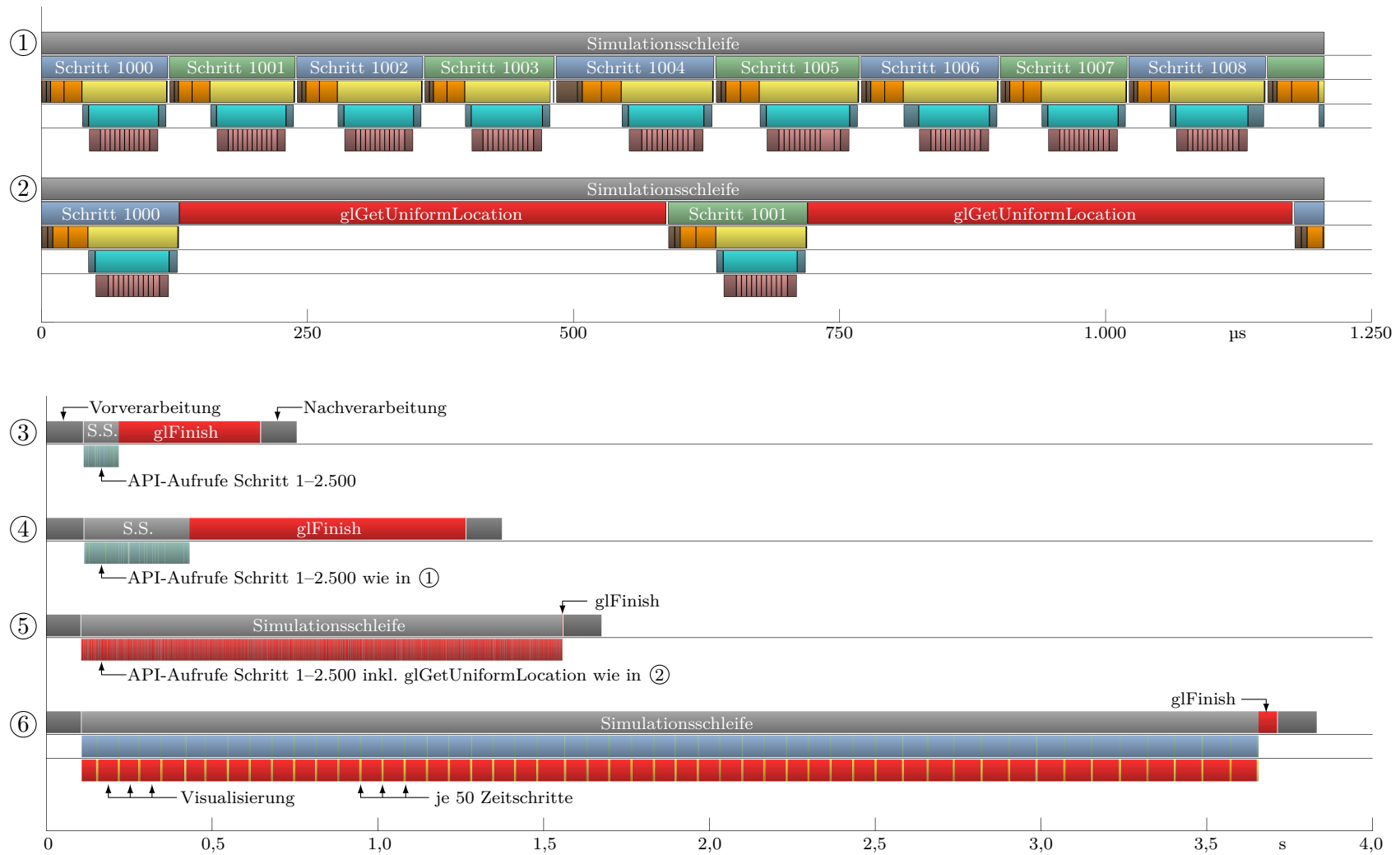


Abbildung 5.11: ①: Normaler Ablauf aufeinanderfolgender Zeitschritte mit Zustands-caching durch das Simulationsframework. ②: Bereits das Einfügen eines einzelnen ungecachten `glGet*`-Befehls reduziert die Anzahl der API-seitig in derselben Zeit durchführbaren Zeitschritte um den Faktor 4–5. ③–⑥: Simulationsverlauf für 2.500 Zeitschritte ohne (③) und mit (④) der Berechnung bahnbbezogener Zielgrößen sowie verzögert durch einen einzelnen `glGet*`-Befehl (⑤) oder durch die Visualisierung in jedem 50sten Zeitschritt (⑥).

kann jedoch nicht auf die Visualisierung verzichtet werden, da dadurch der Ablagerungsprozess besser nachvollzogen werden kann. Weitergehende Laufzeitmessungen in Bezug auf verschiedene Parametrisierungen und Eingabegrößen der Simulation werden in Kapitel 6.2 behandelt.

5.6 Einbindung einer thermodynamischen Simulation

Die Simulation des Temperaturverlaufs im Bauteil während des Beschichtungsprozesses ist ein wichtiges Hilfsmittel zur Analyse des Prozesses, da dieser einen großen Einfluss auf die Qualität der Beschichtung und des Bauteils hat. Insbesondere bestimmt die thermische Belastung den resultierenden Eigenspannungszustand von Beschichtung und Bauteil und damit die potentielle Haltbarkeit der Beschichtung sowie die resultierende Bauteil deformation. Die Realisierung und Integration einer thermodynamischen Simulation selbst ist nicht Gegenstand dieser Arbeit. Das entwickelte Simulationsframework ermöglicht jedoch die einfache Einbindung einer solchen Simulation, falls diese auf einer tetraederbasierten Vernetzung der Bauteilgeometrie basiert, deren Randnetz das für die Beschichtungssimulation verwendete Dreiecksnetz ist. In diesem Fall kann, neben dem Footprintprofil, welches die Massenflusscharakteristik des Prozesses definiert, auch ein thermischer Footprint des Prozesses definiert werden. Dieser thermische Footprint wird in demselben Koordinatensystem wie der Massenfluss-Footprint definiert und gibt die durch den Brenner richtungsabhängig auf der Bauteiloberfläche induzierte Randtemperatur an, welche für jeden Vertex ähnlich wie der Schichtauftrag jedoch ohne eine Akkumulation in jedem Zeitschritt berechnet wird. Des Weiteren können auch die in jedem Zeitschritt an den Vertices abgelagerten Schichtvolumina zur Ableitung der durch den Transport des heißen Schichtmaterials zur Bauteiloberfläche eingebrachten thermischen Energie genutzt werden. Die berechneten Temperaturen und Schichtvolumina stellen also die Eingangsrandbedingungen für eine thermodynamische Simulation dar, welche mittels der Beschichtungssimulation auch für komplexe Bauteilgeometrien auf einfache Weise berechnet werden können.

Wurden mit einem Zeitschritt der Beschichtungssimulation auch die Bauteiltemperaturen neu berechnet, werden diese GPU-seitig gespeichert und in die zuvor beschriebene Berechnung der bahnbezogenen Zielgrößen eingebunden. Dabei werden jeweils die durchschnittliche, minimale und maximale Temperatur des Bauteils berechnet. In Kooperation mit dem Institut für Mechanik der TU Dortmund wurde eine entsprechende Simulationskomponente entwickelt und in [10] dokumentiert.

5.7 Visualisierung

Die Visualisierung der berechneten Zielgrößen ermöglicht die manuelle Inspektion eines simulierten Experiments zur Identifikation und Analyse von Abweichungen und zur Veranschaulichung der geometrischen Zusammenhänge. Die Visualisierung der berechneten oberflächenbe-

zogenen Zielgrößen erfolgt als Einfärbung des mittels OpenGL gerenderten dreidimensionalen Bauteils, wobei unterschiedliche Farbskalen gewählt werden können. Darüber hinaus kann die Beschichtung der Oberfläche durch die Verschiebung der Oberflächenvertices räumlich dargestellt werden, wobei eine Überhöhung des Profils um einen einstellbaren Faktor durchgeführt wird, um auch dünne Schichten visuell erfassen zu können. Ebenso wie die Simulation erfolgt diese Visualisierung ohne Transfer der darzustellenden Daten zur CPU direkt auf Basis der GPU-Datenstrukturen. Dies ist insbesondere für eine Live-Ansicht während der Simulation entscheidend. Zugunsten einer besseren räumlichen Wahrnehmung der Darstellung kann auf Kosten der Exaktheit der Farbskaladarstellung ein Cook-Torrance-Shader für die Schattierung des Werkstücks aktiviert werden [31]. Nach Beenden der Simulation können die oberflächenbezogenen Zielgrößen ferner für Visualisierungen in anderen Programmen exportiert sowie Schnittgraphen entlang beliebiger Schnittebenen durch das Werkstück berechnet und dargestellt werden.

Für die bahnbezogenen Zielgrößen wurden drei unterschiedliche Visualisierungen integriert. Eine Live-Vorschau der berechneten Werte während der Simulation kann als zweidimensionales Liniendiagramm eingeblendet oder als Einfärbung der dreidimensionalen Roboterbahn dargestellt werden. Beide Methoden arbeiten auch hier aus Effizienzgründen direkt auf den GPU-Datenstrukturen, wobei die Liniendiagramme durch einen Geometry-Shader erzeugt werden. Des Weiteren steht nach der Simulation und dem Transfer der Ergebnisse aus dem Grafikspeicher eine komfortablere, frei zoom- und navigierbare Umgebung zur Betrachtung und Auswertung der Zielgrößen in Form von Diagrammen über der Simulationszeit zur Verfügung. Durch Auswählen einzelner Punkte oder ganzer Intervalle in dieser Diagramme wird der Spritzbrenner in der 3D-Ansicht an der entsprechenden Stelle angezeigt und nicht ausgewählte Teile der Roboterbahn werden ausgeblendet. Dadurch kann ein direkter Bezug einzelner Datenpunkte und -intervalle zum Bahnverlauf und der lokalen Spritzkegelgeometrie hergestellt werden. Zahlreiche Darstellungen der Visualisierungen finden sich in Kapitel 6.

5.8 Kalibrier- und Analysemodule

In diesem Abschnitt werden zwei Module des Simulationsframeworks vorgestellt, welche die implementierte Simulation nutzen, um wichtige Simulationsparameter anhand gefertigter und vermessener Schichtdickenverteilungen zu kalibrieren und eine Sensitivitätsanalyse durchzuführen. Auf Basis bereits erfolgreich kalibrierter Simulationsparameter kann die Funktionalität der Parameterkalibrierung weiterhin dazu verwendet werden, im Fertigungsprozess eines Bauteils abweichende Parameter zu identifizieren. Beide Module, die Parameterkalibrierung und die Sensitivitätsanalyse, profitieren, wie auch eine extern angekoppelte Bahnplanungssoftware, von der schnellen Ausführung einer Vielzahl von unterschiedlich parametrisierten Simulationen.

5.8.1 Automatische Parameterkalibrierung

Systematische Abweichungen zwischen einer gemessenen Schichtdickenverteilung eines gefertigten Bauteils und der von der Beschichtungssimulation berechneten Schichtdickenverteilung, also Abweichungen, die nicht auf Prozess- oder Messrauschen zurückzuführen sind, deuten auf eine nicht exakt kalibrierte Simulation oder eine fehlerhafte Parametrisierung des Fertigungsprozesses für das Bauteil hin. Unter der Voraussetzung, dass das verwendete Beschichtungsmodell den Prozess abbilden kann, bietet die automatische Parameterkalibrierung die Möglichkeit, eine Parameterbelegung für wichtige Simulationsparameter zu ermitteln, welche – im Sinne einer Minimierung des quadratischen Schichtfehlers – zu einer möglichst optimalen Reproduktion der gemessenen Schichtdickenverteilung durch die Simulation führen. Folgende Parameter der Simulation können dabei optimiert werden:

- Ein lateraler Offset $\Delta \mathbf{x}_{\mathcal{F}} = (\Delta x_{\mathcal{F}}, \Delta y_{\mathcal{F}})$ des Footprints, welcher das verwendete Footprintprofil orthogonal zur Spritzrichtung im Footprintkoordinatensystem verschiebt. Standardwert ist $(\Delta x_{\mathcal{F}}, \Delta y_{\mathcal{F}}) = (0, 0)$.
- Ein Skalierungsfaktor $s_{\mathcal{F}}$, welcher die Höhe des Footprintprofils skaliert und somit den Gesamtmassenfluss steuert. Standardwert ist $s_{\mathcal{F}} = 1$.
- Die Stützstellen $(\theta_i, \xi_{\text{ang},i})$ der winkelabhängigen Haftwirkungsgradfunktion ξ_{ang} . Die Ausgangswerte zur Kalibrierung des Haftwirkungsgrades werden gemäß Gleichung (4.31) gesetzt. Als Standardwert dienen im Weiteren die Ergebnisse einer initialen Parameterkalibrierung des Haftwirkungsgrades (vergleiche Kapitel 4.4.4).
- Eine Verschiebung des Werkstücks um $\Delta \mathbf{x}_{\mathcal{W}} = (\Delta x, \Delta y, \Delta z)_{\mathcal{W}}$. Standardwert ist $\Delta \mathbf{x}_{\mathcal{W}} = (0, 0, 0)$.
- Eine Rotation des Werkstücks um die Eulerwinkel $(\gamma_z, \gamma_{y'}, \gamma_{x''})$ um die in den Schwerpunkt des Werkstücks verschobenen Achsen des Werkstückkoordinatensystems. Die Rotation erfolgt hier wahlfrei entsprechend der z, y', x'' -Rotationskonvention zuerst um die z -Achse, gefolgt von der rotierten Achse y' und der zweimal rotierten Achse x'' . Standardwert ist auch hier $(0, 0, 0)$.

Während eine automatische Kalibrierung des Skalierungsfaktors $s_{\mathcal{F}}$ sowie der Stützstellen des Haftwirkungsgrades in der Regel zur initialen Kalibrierung der Simulation verwendet wird (vergleiche Kapitel 4.4.3), dient die Bestimmung optimaler Werkstückverschiebungen und -rotationen in erster Linie der Feststellung von Abweichungen zwischen experimentellem Versuchsaufbau und der Modellierung in der Simulation. Das Auftreten eines nicht zu vernachlässigenden lateralen Footprintoffsets $\Delta \mathbf{x}_{\mathcal{F}}$ kann sowohl auf eine Fehleinmessung des Footprints als auch auf eine unpräzise Montage des Spritzbrenners auf dem Roboter oder eine abweichende Ausrichtung des Spritzstrahls hindeuten. Ferner ist zu beachten, dass die Orthogonalität verschiedener optimierbarer Parameter von der Bauteilgeometrie abhängt. So kann beispielsweise ein lateraler Footprintoffset bei der Beschichtung einer ebenen Probe mit

einer Linearbahn in der Regel ebenso durch eine Werkstückverschiebung kompensiert werden. Wird jedoch eine Kugel mit einer helixförmigen Roboterbahn beschichtet (vergleiche Abbildung 5.9), ist eine solche Kompensation im Allgemeinen nicht möglich.

Die Optimierung der ausgewählten Parameter für einen in der Simulationsumgebung modellierten Spritzversuch erfolgt mittels des Gauß-Newton-Verfahrens zur Lösung nichtlinearer Minimierungsprobleme basierend auf einer Least-Squares-Formulierung [50]. Zu diesem Zweck werden die zu optimierenden Parameter formal in einen Vektor $\mathbf{a} = a_i$ für $i = 1, \dots, p$ zusammengefasst. Es bezeichne $s(\mathbf{a}) \in \mathbb{R}^m$ die von der Simulation für $m > p$ Messpunkte berechnete und $\mathbf{y} \in \mathbb{R}^m$ die an den gleichen Punkten gemessene Schichtdickenverteilung. Damit ergibt sich für das Residuum $\mathbf{r}(\mathbf{a}) = s(\mathbf{a}) - \mathbf{y}$ der quadratische Schichtfehler

$$f(\mathbf{a}) = \frac{1}{2} \sum_{i=1}^m r_i(\mathbf{a})^2 = \frac{1}{2} \|\mathbf{r}(\mathbf{a})\|_2^2.$$

Für die Minimierung

$$\min_{\mathbf{a}} f(\mathbf{a})$$

des Schichtfehlers wird die Iterationsvorschrift des Gauß-Newton-Verfahrens

$$\mathbf{a}_{n+1} = \mathbf{a}_n - [\mathbf{J}_{\mathbf{r}}^T \mathbf{J}_{\mathbf{r}}]^{-1} \mathbf{J}_{\mathbf{r}}^T \mathbf{r}. \quad (5.5)$$

verwendet, welche sich aus dem Newtonverfahren durch die für die Minimierung quadratischer Fehlersummen geeignete Approximation der Hessematrix mittels $\mathbf{H}_f \approx \mathbf{J}_{\mathbf{r}}^T \mathbf{J}_{\mathbf{r}}$ ergibt [50]. Die Berechnung der für die Jacobimatrix $\mathbf{J}_{\mathbf{r}}$ benötigten partiellen Ableitungen nach den Simulationsparametern a_1, \dots, a_p werden dabei als numerische Differenzenquotienten durch $(p+1)$ -fache Ausführung der Simulation berechnet. Die Iteration (5.5) wird abgebrochen, sobald sich die Parameter \mathbf{a}_n von Iteration zu Iteration nicht mehr wesentlich verändern oder eine vorgegebene maximale Iterationsanzahl erreicht wird.

Um eine Optimierung basierend auf mehreren Versuchsergebnissen durchzuführen, können mehrere Bauteilmodelle mit zugehörigen Roboterbahnen und unterschiedlichen Zielbeschichtungen simultan optimiert werden. Die Zielvektoren \mathbf{y} sowie die Simulationsergebnisse $s(\mathbf{a})$ und Residuen \mathbf{r} unterschiedlicher Simulationen werden dabei in je einem Vektor zusammengefasst. Für die Bestimmung des Haftwirkungsgrades wurden beispielsweise sieben unterschiedliche Bahnen und Zielgrößen für die Beschichtung der Flachproben aus unterschiedlichen Winkeln einbezogen (Kapitel 4.4.4).

Während die Optimierung des Haftwirkungsgrades aufgrund der einfachen Probengeometrien und der dediziert für die Bestimmung der einzelnen Parameter entwickelten Experimente äußerst robust ist, ist dies bei komplexeren Bauteilgeometrien keineswegs garantiert. So zeigte sich für das Bauteil MAT3D beispielsweise eine deutlich komplexere Optimierungslandschaft und eine erhöhte Abhängigkeit vom Startpunkt für eine Optimierung mehrerer Parameter mit wechselseitigen Abhängigkeiten. Für diesen Fall wurden Methoden integriert, um den p -dimensionalen Parameterraum in einem gegebenen Intervall mit p gegebenen Schrittweiten vollständig abzutasten und verschiedene Fehlerwerte zu berechnen.

5.8.2 Sensitivitätsanalyse

Das implementierte Modul zur Sensitivitätsanalyse bietet die Möglichkeit, die Sensitivität der Beschichtung eines Bauteils gegenüber Parameterveränderungen zu untersuchen. Das Ziel ist dabei insbesondere, die Empfindlichkeit eines Fertigungsaufbaus gegenüber einer Fehleinmessung oder -platzierung des Bauteils oder des Spritzbrenners zu ermitteln und so bereits vor einer Fertigung kritische Parameter zu identifizieren. Diese sollten beim Aufbau und beim Einmessen der Fertigungsumgebung besonders beachtet werden.

Die Grundidee besteht dabei darin, eine automatisierte Sensitivitätsanalyse durchzuführen, bei der jeweils einer oder mehrere der im vorherigen Kapitel aufgezählten Parameter variiert und Abweichungen der aus der Simulation resultierenden Beschichtung anhand von Fehlermaßen quantifiziert werden. Hierfür werden zunächst die zu untersuchenden Parameter ausgewählt und für jeden das zu untersuchende Parameterintervall sowie die Schrittweite zur Abtastung dieses Intervalls festgelegt. Anschließend werden die Simulationen für die angegebenen Parametrisierungen automatisiert durchgeführt, wobei entweder nur jeweils ein Parameter gleichzeitig variiert wird („One-At-A-Time-Analyse“) oder ein vollfaktorielles Schema verwendet wird, um jede Kombination der Parameterwerte zu untersuchen. Für jede durchgeführte Simulation wird die resultierende Schichtdickenverteilung sowie die Abweichungen von der Schichtdickenverteilung der gewählten Basisparametrisierung gespeichert. Diese Abweichungen werden in Form des mittleren und maximalen Absolutfehlers, des summierten quadratischen Fehlers (engl.: Sum of Squared Errors, SSE) sowie des RMSE (engl.: Root-Mean-Square Error) ermittelt.

Eine oberflächenbezogene Visualisierung der resultierenden Abweichungen gegenüber der Basisparametrisierung oder zwischen zwei auswählbaren Parametrisierungen kann als Falschfarbdarstellung in der 3D-Ansicht des Bauteils sowie als Fehlerhistogramm angezeigt werden, wobei eine definierte „Region of Interest“ berücksichtigt wird. Im Fall einer One-At-A-Time-Analyse werden zusätzlich die Fehlerwerte in Abhängigkeit der variierten Parameter als Kurvendiagramm dargestellt. Beispiele für diese Darstellungen sind im Rahmen der Sensitivitätsanalyse für die Bauteile MAT2A und MAT3D in Kapitel 6.4.1 zu finden.

KAPITEL

Evaluation

Zur Evaluation des entwickelten Simulationssystems sowie des Beschichtungsmodells, welches in Kapitel 4 beschrieben wird, wurden eine Reihe von Simulationsläufen durchgeführt, deren Ergebnisse in diesem Kapitel vorgestellt werden.

Im folgenden Abschnitt 6.1 werden zunächst die Ergebnisse einiger Simulationsläufe unter Verwendung verschiedener synthetisch generierter und nicht real gefertigter Bauteilgeometrien vorgestellt, um die Implementierung für einfache Testfälle zu verifizieren und einen anschaulichen Überblick über die berechneten Zielgrößen zu geben. Die Interpretation der Zielgrößen anhand dieser einfachen synthetischen Bauteile erleichtert die Einschätzung dieser Zielgrößen für reale Bauteile, bei denen komplexe geometrische Zusammenhänge nicht immer intuitiv erkennbar sind. Ferner wird in diesem Abschnitt der Einfluss der Zeitschrittweite und der gewählten Tiefenpufferauflösung betrachtet. Ebenfalls anhand synthetischer Bauteile werden anschließend in Kapitel 6.2 die erreichbaren Rechenzeiten der Simulation für Eingaben mit unterschiedlichen Messpunktanzahlen untersucht, welche von zentraler Bedeutung für die Einsetzbarkeit der Simulation in gekoppelten Bahnplanungs- und Bahnoptimierungsprozessen ist.

In Kapitel 6.3 wird die Simulation anhand der Beschichtung zweier gefertigter Segmente eines komplexen Tiefziehwerkzeugs evaluiert. Gegenstand dieses Kapitels sind die Fertigung und Vermessung der Beschichtung, der Vergleich mit den simulierten Schichtdicken sowie die Darstellung der sekundären oberflächen- und bahnbezogenen Zielgrößen. Im abschließenden Kapitel 6.4 wird die Nutzung der Simulation zur Sensitivitätsanalyse sowie zur Bahnoptimierung anhand von Beispielen demonstriert.

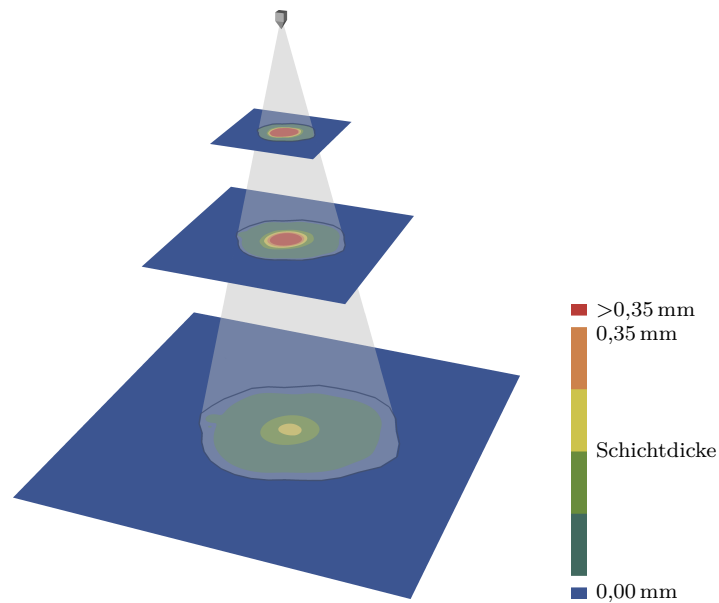


Abbildung 6.1: Schichtdicke im simulierten Footprintversuch mit drei in unterschiedlichen Spritzabständen 55 mm, 110 mm und 220 mm platzierten virtuellen Substratebenen. Die im Standardspritzabstand $d_{\mathcal{F}} = 110$ mm berechnete Schichtdickenverteilung entspricht bis auf minimale Gleitkommatauglichkeiten dem Profil des verwendeten Footprints. Für größere Abstände sinkt die berechnete Profilhöhe erwartungsgemäß und für kürzere Distanzen steigt sie.

6.1 Demonstration und Verifikation grundlegender Zusammenhänge

Eine umfassende Verifikation der Simulationsimplementierung auf formaler Basis ist für die entwickelte Simulation kaum durchführbar. Allerdings können einfache Simulationsversuche aufzeigen, dass die Implementierung in diesen Fällen korrekt funktioniert und dem in Kapitel 4 aufgestellten Modell genügt. Gleichzeitig veranschaulichen diese Beispiele die grundlegenden Zusammenhänge zwischen den geometrischen Parametern und den berechneten Zielgrößen.

6.1.1 Virtueller Footprintversuch

Die virtuelle Nachbildung eines Footprintversuchs bietet zunächst die einfachste Möglichkeit, das Simulationsergebnis zu überprüfen. Unter Verwendung eines ebenen Bauteilmodells mit Messpunkten, welche wie in der verwendeten Footprinttextur in einem Gitter mit Gitterabstand 0,125 mm angeordnet sind, sowie einer Roboterbahn, bei der die Spritzpistole für die Footprintspritzdauer an einer fixen Position zentriert über dem Bauteil steht, kann der Footprintversuch simulativ nachgestellt werden. Die berechnete Schichtdickenverteilung muss in diesem Fall mit der des Footprints selbst übereinstimmen. Abbildung 6.1 zeigt das Ergebnis in der Simulationsumgebung, wobei das ebene Bauteil im Spritzabstand 110 mm des Footprints sowie in den Abständen 55 mm und 220 mm platziert wurde. Alle Ebenen sind dabei so groß, dass sie den simulierten Spritzstrahlquerschnitt ausfüllen. Die Sichtbarkeits-

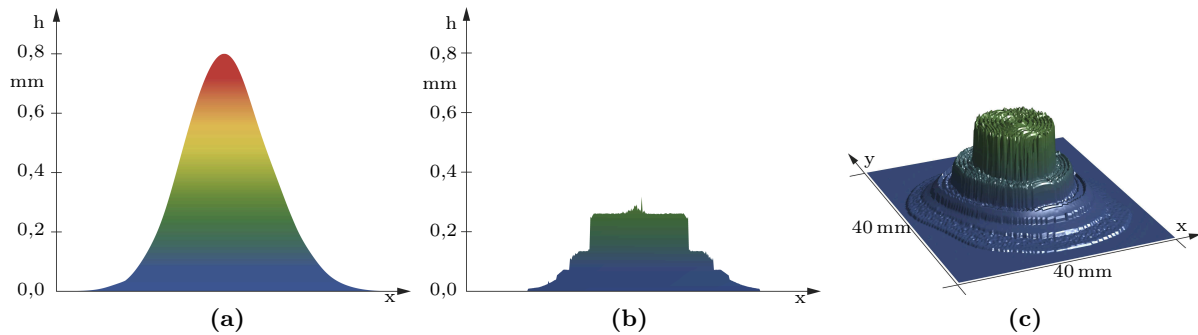


Abbildung 6.2: Durch die Wahl eines zu geringen Zeitschritts können numerische Probleme bei der Akkumulation zu vieler sehr kleiner Schichtdickeninkremente auftreten. (a) Abgelagerte Schichtdicke h im virtuellen Footprintversuch mit $\delta t = 1 \cdot 10^{-3}$ s ohne Beeinträchtigungen. (b), (c) Ergebnis desselben Versuchs für $\delta t = 2 \cdot 10^{-4}$. Die zu kleinen Schichtdickeninkremente sind nicht mehr korrekt auf die akkumulierten Schichtdicken addierbar.

berechnung wurde deaktiviert, um die Schichtablagerung auf allen drei Ebenen gleichzeitig zu simulieren. Der Haftwirkungsgrad wurde konstant auf Eins gesetzt. Wie erwartet wird bei der mittleren Entfernung $d_{\mathcal{F}} = 110$ mm das ursprüngliche Footprintprofil abgelagert, wobei der durchschnittliche absolute Fehler in der Footprintfläche ca. $0,4 \mu\text{m}$ beträgt. Dieser Fehler ist auf Gleitkommaraunauigkeiten bei der Auswertung des Shader-basierten Modells zurückzuführen. Die Tatsache, dass dieser Fehler auch bei der Variation der Anzahl an Zeitschritten im Bereich von einem Zeitschritt bis zu 1.000 nahezu konstant bleibt, zeigt, dass die Implementierung der Zeitschrittroutine und der Berechnung des zeitlichen Ablagerungsfaktors $t_{\mathcal{F}}$ korrekt funktioniert.

Einfluss des gewählten Zeitschritts Δt

Der Simulationszeitschritt Δt kann jedoch nicht beliebig klein gewählt werden. Anhand des virtuellen Footprintversuchs lässt sich dies gut nachvollziehen, da hier an jedem Messpunkt aufgrund der stationären Spritzpistole in jedem Schritt dasselbe Schichtvolumen abgelagert wird. Eine weitere Verringerung des Zeitschritts ausgehend von $1 \cdot 10^{-3}$ Sekunden führt hier im Bereich zwischen $2 \cdot 10^{-4}$ und $4 \cdot 10^{-4}$ Sekunden zu numerischen Instabilitäten, bei denen die pro Zeitschritt abzulagernden Schichtdicken und -volumina so klein werden, dass sie bei der Summierung der Schichtdicke die relative Gleitkommarechengenauigkeit unterschreiten. Dieses bereits in Kapitel 5.5.1 diskutierte Verhalten zeigt sich im Ergebnis darin, dass die berechneten Schichtdicken an den einzelnen Messpunkten nicht mehr weiter ansteigen, da die zu addierenden Terme so klein sind, dass sie keine Änderung mehr bewirken. Wie in Abbildung 6.2b und 6.2c für einen Zeitschritt von $2 \cdot 10^{-4}$ s zu erkennen ist, stagniert das Profilwachstum im virtuellen Footprintversuch an den Messpunkten an unterschiedlichen Schichtdickengrenzwerten. Dies ist darauf zurückzuführen, dass für die größeren Summanden in der Mitte des Footprints die Grenze der relativen Gleitkommagenauigkeit erst bei einer höheren Schichtdicke erreicht wird als für die kleineren Summanden im Randbereich des Footprints.

Erkennbar sind derartige numerische Probleme bei Simulationsläufen an einer deutlichen Abnahme der Prozesseffizienz sowie unter Umständen der Anzahl pro Zeitschritt beschichteter Vertices, obwohl der Brenner durchgehend auf eine Bauteiloberfläche mit näherungsweise gleichverteilten Messpunkten gerichtet ist. Diese Werte werden als bahnbezogene Zielgrößen berechnet und dargestellt. In der zu dem in Abbildung 6.2b dargestellten Ergebnis führenden Simulation bleibt die Anzahl pro Zeitschritt beschichteter Messpunkte beispielsweise zunächst für ca. 1.600 Zeitschritte konstant bei 32.010 und fällt dann stufenweise auf ungefähr 600 Messpunkte am Ende der einsekündigen Simulationsdauer ab. Demgegenüber ist bei $\Delta t = 1 \cdot 10^{-3} \text{ s}$ kein Absinken feststellbar (Abbildung 6.2a).

Bei Simulationen mit üblichen, nicht-stationären Roboterbahnen wird diese unvermeidbare numerische Problematik durch die an jedem Messpunkt in jedem Zeitschritt wechselnden Summanden unterschiedlicher Größenordnung teilweise verdeckt und erst bei deutlich kleineren Zeitschritten sichtbar. Insbesondere bei Beschichtungssimulationen mit vielen Beschichtungsüberläufen, sehr kleinen Zeitschritten oder allgemein geringer Ablagerungsleistung kann es aber auch hier zu leicht sichtbaren Einbrüchen der berechneten Prozesseffizienz oder der Anzahl beschichteter Vertices kommen, welche daher für jeden grundlegend neuen Simulationslauf kritisch geprüft werden sollten. Unproblematisch ist hingegen in der Regel die typische Abnahme der Anzahl der beschichteten Messpunkte pro Zeitschritt gleich zu Beginn einer Simulation, welche auf die üblicherweise große Anzahl der extrem kleinen Profilwerte $\mathcal{F}(x, y) < 10^{-10} \text{ mm}$ im Randbereich des Footprints zurückzuführen ist. Diese Werte sind bereits auf Schichtdicken von mehr als $0,1 \mu\text{m}$ nicht mehr im Rahmen der einfachen Gleitkommagenauigkeit addierbar und tragen insgesamt so wenig zum Gesamtvolumenstrom bei, dass die Prozesseffizienz im Gegensatz zur Anzahl beschichteter Messpunkte nur in zu vernachlässigendem Rahmen sinkt. Des Weiteren ist bei der Interpretation der beiden bahnbezogenen Größen zu beachten, dass numerische Probleme auch durch eine zu feine Bauteilvernetzung hervorgerufen werden können. In diesem Fall kann es passieren, dass die Schichtdicken zwar korrekt und stabil berechnet werden, jedoch die für die Prozesseffizienzberechnung verwendeten Volumina an den Messpunkten zu klein werden. Dies ist durch eine gröbere Vernetzung oder größere Zeitschritte vermeidbar.

Abschließend bleibt anzumerken, dass derartige numerische Probleme bei den im Rahmen dieser Arbeit durchgeführten Simulationen auch für sehr fein vernetzte Bauteile nicht auftraten und derartige Probleme für übliche Vernetzungen und Zeitschritte nicht zu erwarten sind.

Einfluss des Spritzabstandes

Anhand der in Abbildung 6.1 gezeigten Beschichtung der drei Ebenen in unterschiedlichen Abständen lässt sich ebenfalls die Entfernungsabhängigkeit des Beschichtungsmodells demonstrieren und grundlegend verifizieren. Die Betrachtung der erzielten Schichtdicken auf den drei Ebenen bestätigt hier, dass die resultierende Schichtdicke bei Halbierung des Spritzabstandes entsprechend des Beschichtungsmodells nach Gleichung (4.28) an jedem Messpunkt mindes-

tens um den Faktor vier ansteigt, wobei dieser Faktor umso größer ist, je höher das Footprintprofil an dieser Stelle ist. Dies liegt daran, dass die Höhe des zur Modellierung verwendeten Kegelstumpfelements bei einer Vergrößerung des (Ablagerungs-)Volumens schneller wächst als das Volumen. Bei einer Verdoppelung des Spritzabstandes sinkt die Schichtdicke entsprechend auf etwas weniger als das 0,25-fache. Das abgelagerte Schichtvolumen bleibt erwartungsgemäß für alle Abstände identisch. Für das zylindrische Beschichtungsmodell nach Gleichung (4.29) liegt keine Spritzstrahldivergenz vor und die Footprintprofile sind dementsprechend bei allen drei Abständen identisch.

6.1.2 Bahnzielgrößen am Beispiel geneigter Ebenen

Zur Überprüfung und Veranschaulichung bahnbezogener Zielgrößen sowie der hierfür benötigten Grundgrößen, Spritzabstand und -winkel, wurde eine Beschichtungssimulation für die in Abbildung 6.3 dargestellten Flächen durchgeführt, welche um 0° , 20° , 40° und 60° zur Spritzrichtung geneigt sind. Dabei wurden die hinteren und vorderen vier Flächen mit je einer übergreifenden Roboterbahn simuliert. Die hinteren vier Flächen in Reihe **A** der Abbildung zeigen zunächst die nach der Simulation der dargestellten Linearbahn resultierende Schichtdickenverteilung, welche mit steigender Neigung der Flächen abnimmt. Die erreichte Prozesseffizienz ist als Einfärbung der mit Spritzrichtungsmarkern dargestellten Spritzdüsenbahn veranschaulicht und nimmt aufgrund der mit dem Spritzwinkel abnehmenden Ablagerungseffizienz (vergleiche Seite 136) von Fläche zu Fläche ab. Für die Bahnabschnitte zwischen den Ebenen, bei denen sich der virtuelle Spritzbrenner nicht im Eingriff befindet, ist sie Null. Die vordere Reihe **B** veranschaulicht den durchschnittlichen Spritzabstand, gemittelt über die gesamte Roboterbahn, unter dem jeder Oberflächenpunkt beschichtet wird. Dabei sind für die geneigten Flächen deutlich die von der Roboterbahn an der Düse ausgehenden kreisförmigen Isokonturen erkennbar.

Die wichtigen Spritzwinkel- und Abstands-basierten, bahnbezogenen Zielgrößen für diese Bahn sind in Abbildungen 6.3b und 6.3c dargestellt, wobei Bahnabschnitte, bei denen sich der Brenner nicht im Eingriff befindet, grau markiert sind. Die Bahn in Abbildung 6.3a wird dabei von links nach rechts abgefahren. Der maximale und der durchschnittliche Spritzwinkel liegen dabei stets in einem knapp 20° breiten Intervall zentriert um die jeweilige Flächenneigung (0 – 60°). Erwartungsgemäß ist dabei der durchschnittliche Spritzwinkel aufgrund der Divergenz des Spritzkegels für jede der Flächen bei Eintritt der Fläche in den Kegel zunächst größer als die Flächenneigung und sinkt bei Austritt der Fläche aus dem Kegel unter den Neigungswinkel der Fläche. Der maximale Spritzwinkel bleibt für jede Fläche zunächst auf dem höheren Niveau und sinkt schließlich mit dem durchschnittlichen Winkel ab. Dass hierbei der durchschnittliche Spritzwinkel abschnittsweise den maximalen Winkel überschreitet, ist darin begründet, dass der maximale Winkel innerhalb eines kleineren Teils des Spritzkegels berechnet wird (vergleiche Kapitel 5.5.1). Im Vergleich mit der Anordnung der Flächenstücke in Abbildung 6.3a erschließt sich des Weiteren unmittelbar die Korrektheit der berechneten abstandsbezogenen Bahngrößen. Auch hier ist der durchschnittliche Spritzabstand nicht

6. EVALUATION

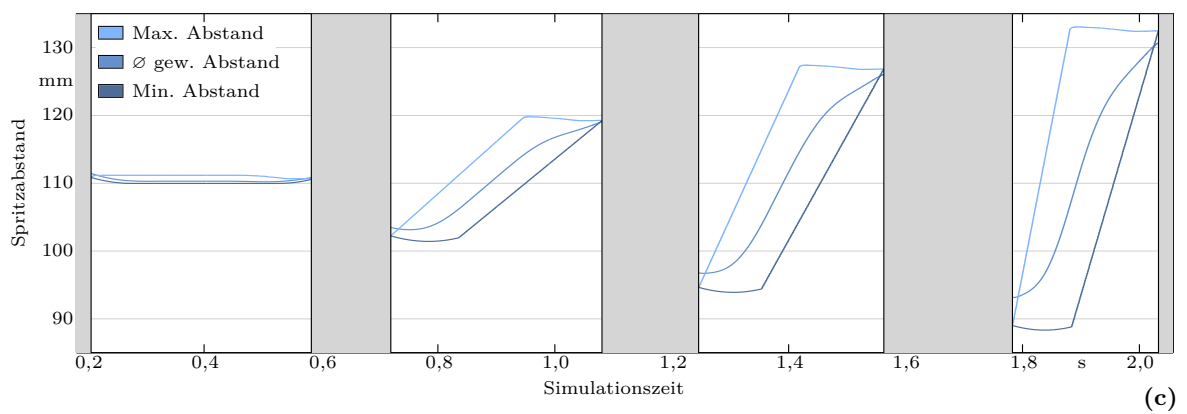
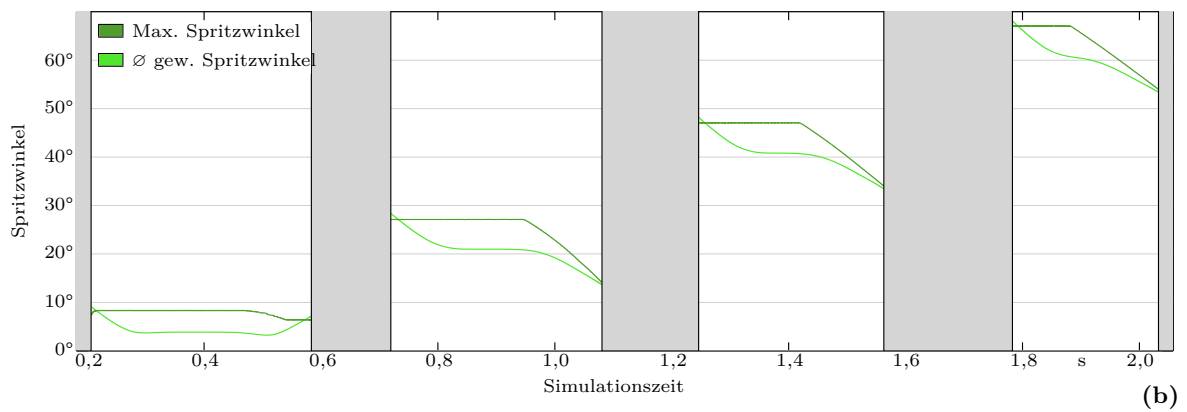
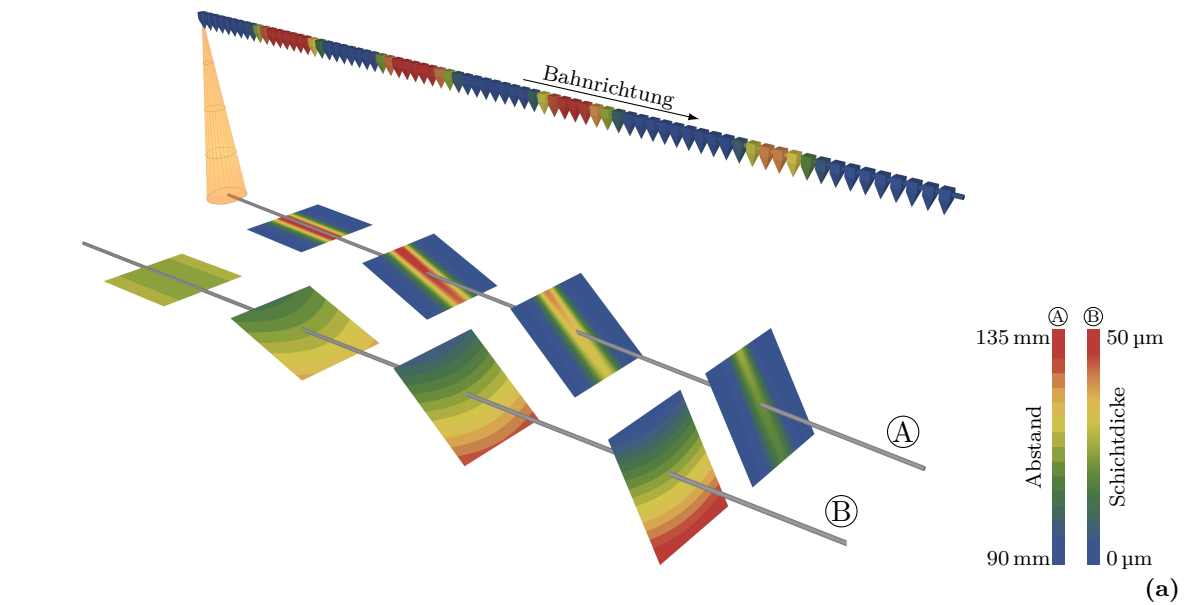


Abbildung 6.3: (a) Beispiel einer Beschichtung von Flächenstücken, welche um 0° , 20° , 40° und 60° zur Bahnrichtung geneigt sind. (b) Darstellung des bahnbezogenen Verlaufs von Spritzwinkeln und (c) Spritzabständen .

immer durch den minimalen oder maximalen Abstand beschränkt, da letzterer nur sinnvoll innerhalb des Kernstrahls ausgewertet werden kann.

6.1.3 Sichtbarkeitsberechnung und Wahl der Tiefenpufferauflösung

Ein wichtiger Parameter, welcher die Genauigkeit der Simulation beeinflussen kann, ist die Auflösung des zur Sichtbarkeitsberechnung verwendeten Tiefenpuffers. Während bei einer zu hoch gewählten Auflösung die Simulationsgeschwindigkeit deutlich abnehmen kann (vergleiche Kapitel 6.2.1), werden bei einer zu geringen Auflösung in der Regel einige Messpunkte fälschlicherweise als sichtbar oder fälschlicherweise als nicht sichtbar klassifiziert. Eine solche Fehlklassifizierung tritt in der Regel dann auf, wenn mehrere projizierte Bauteilvertices in der Bildebene so dicht beieinander liegen, dass sie auf dasselbe Tiefenfragment abgebildet werden. Besonders anfällig für Fehlklassifizierungen sind daher Teile der Oberfläche, welche unter sehr flachem Winkel betrachtet werden.

Da in dem typischen Anwendungsszenario der Bahnplanung oder -optimierung viele Simulationen für ein Bauteil unter Verwendung ähnlicher Bahnen und Parametrisierungen durchgeführt werden, ist es sinnvoll, eine geeignete Auflösung im Vorhinein zu bestimmen. Zu diesem Zweck und um die Auswirkungen einer zu niedrig gewählten Tiefenpufferauflösung auf die berechnete Beschichtung zu untersuchen, wurde in die Simulationsumgebung eine Funktion integriert, welche automatisch Simulationen mit verschiedenen Tiefenpufferauflösungen durchführt und dabei die Abweichungen der Beschichtungsergebnisse jeder Auflösung gegenüber dem Ergebnis mit maximaler Tiefenpufferauflösung berechnet. Überprüft werden dabei die Auflösungen R , deren Kantenlängen Zweierpotenzen entsprechen, beginnend bei $R = 16^2$ bis zur maximalen Auflösung von $R_{\max} = 16.384^2$ auf dem Testsystem. Als Fehlerkriterien werden folgende Werte jeweils für die Oberfläche innerhalb der definierten Region of Interest, für die Oberflächen außerhalb der Region of Interest sowie für das gesamte Bauteil berechnet und ausgegeben:

- Der prozentuale, von der Auflösung R abhängige Schichtvolumenfehler

$$E_{\text{vol}}(R) = 100\% \cdot \left(\sum_i |V_i(R) - V_i(R_{\max})| \right) / \left(\sum_i V_i(R_{\max}) \right),$$

wobei $V_i(R)$ das an Messpunkt i unter Verwendung der Auflösung R abgelagerte Schichtvolumen darstellt.

- Die maximale Abweichung der Schichtdicke

$$E_{h,\max}(R) = \max_i |h_i(R) - h_i(R_{\max})|,$$

wobei $h_i(R)$ die an Messpunkt i unter Verwendung der Auflösung R abgelagerte Schichtdicke darstellt.

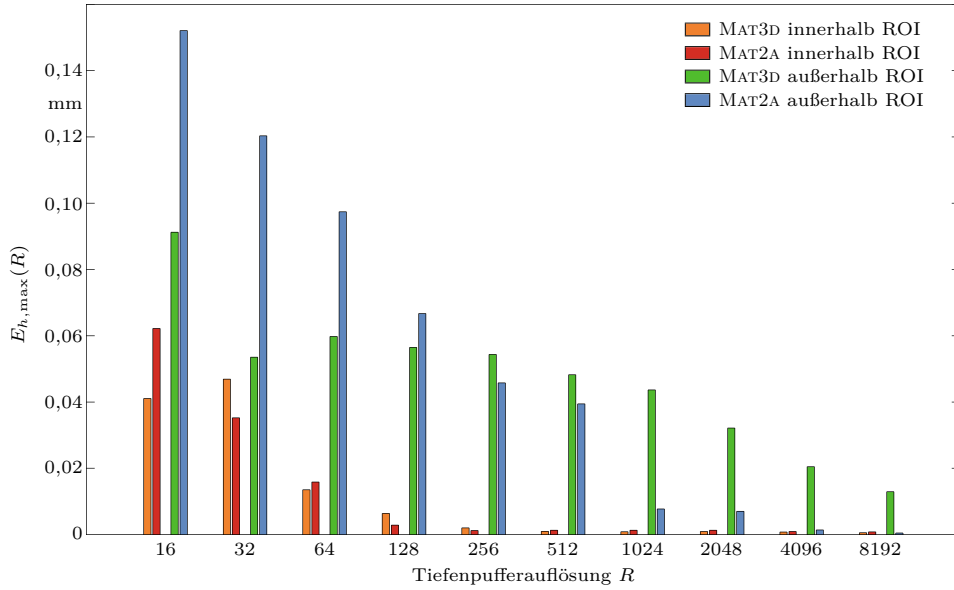


Abbildung 6.4: Maximaler Schichtfehler $E_{h,max}(R)$ für verschiedene Tiefenpufferauflösungen für die Bauteile MAT2A und MAT3D innerhalb und außerhalb der Region of Interest.

Einfluss der Auflösung für verschiedene Bauteile

Für die im Rahmen dieser Arbeit beschichteten Matrizenbauteile MAT3D und MAT2A, welche in Anhang D gezeigt werden, sind die Fehlerwerte $E_{h,max}(R)$ innerhalb und außerhalb der Region of Interest, also der zu beschichtenden Funktionsoberfläche, in Abbildung 6.4 dargestellt. Dabei ist zunächst zu erkennen, dass $E_{h,max}$ erwartungsgemäß mit steigender Auflösung des Tiefenpuffers in der Tendenz abnimmt. Lokal kann es jedoch auch zu Anstiegen kommen, da sich mit der Tiefenpufferauflösung auch die Abtastpositionen der Tiefenfragmente ändern. Des Weiteren ist die maximale Schichtdickenabweichung außerhalb der Region of Interest bei beiden Bauteilen deutlich höher. Dies ist darauf zurückzuführen, dass die Bereiche innerhalb der Region of Interest in der Regel aus näherungsweise senkrechter Anstellung beschichtet werden. Im Gegensatz dazu werden die nicht in der Region of Interest enthaltenen Seitenflächen des Bauteils, wie in Abbildung 6.16f auf Seite 219 zu erkennen ist, unter sehr flachen Anstellwinkeln beschichtet und weisen daher deutlich höhere Abweichungen $E_{h,max}(R)$ auf. Insgesamt reicht für beide Matrizenbauteile in Bezug auf $E_{h,max}$ die Verwendung einer Auflösung zwischen 256^2 und 1024^2 aus, wobei nach Abbildung 6.4 innerhalb der Region of Interest ein maximaler Schichtfehler von 1–2 μm erwartet werden kann. Der durchschnittliche Schichtdickenfehler ist für diese Konfigurationen deutlich geringer und liegt im Bereich von 0,1–0,2 μm sowohl innerhalb als auch außerhalb der Region of Interest.

Da die Matrizenbauteile zwar einige – von den Roboterbahnpositionen aus gesehen – verdeckte Teilflächen enthalten, jedoch fast keine Selbstverdeckung der Geometrie aufweisen, wurde das in Abbildung 6.5a dargestellte filigrane und hochauflösende Fahrradmodell als zusätzliche Testgeometrie verwendet. Speziell für die Untersuchung und Demonstration des Einflusses der Tiefenpufferauflösung wurde – von der Mäanderbahn aus gesehen – hinter dem Fahr-

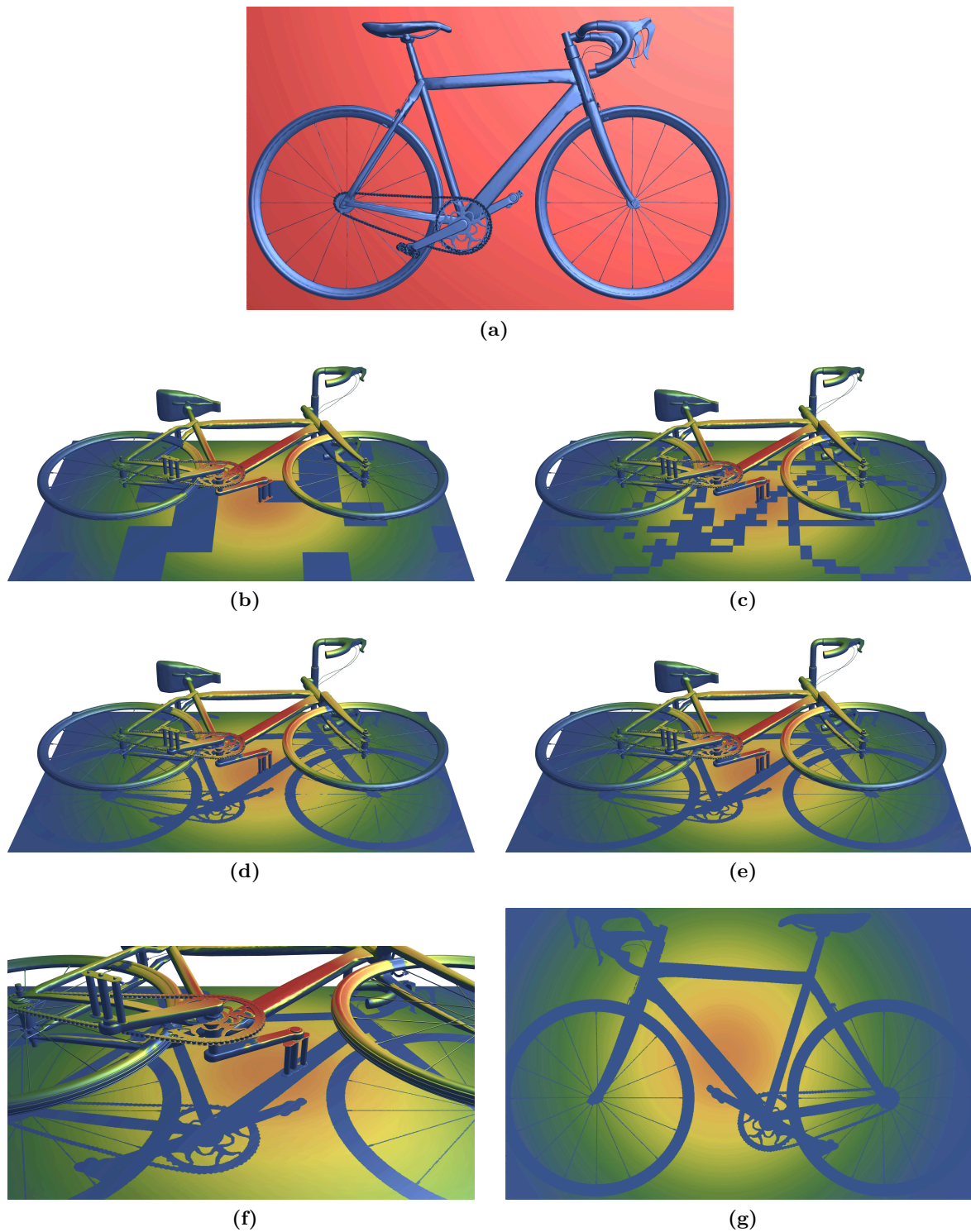


Abbildung 6.5: Veranschaulichung des Einflusses der für die Beschichtungssimulation gewählten Tiefenpufferauflösung anhand des Schattenwurfs eines filigranen Fahrradmodells. Niedrigere Auflösungen verursachen Blockartefakte im Schattenwurf. (a) Fahrradmodell aus Sicht des Spritzbrenners. (b) Tiefenpufferauflösung $R = 16^2$. (c) $R = 64^2$. (d) $R = 1.024^2$. (e) $R = 16.384^2$. Für $R = 16.384^2$ sind auch in der Detailansicht (f) sowie auf der Rückseite der Schattenebene (g) praktisch keine Artefakte mehr erkennbar.

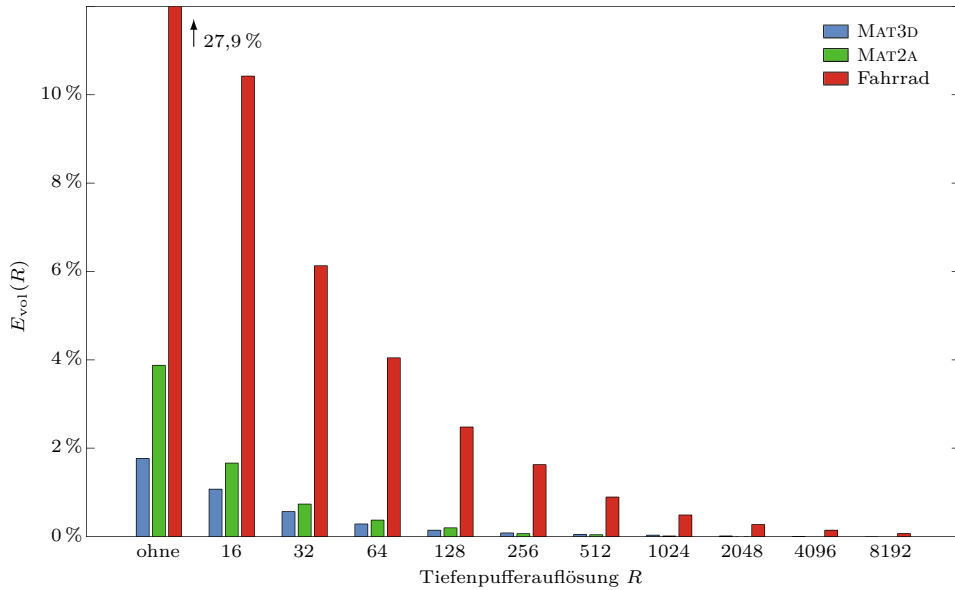


Abbildung 6.6: Durchschnittlicher prozentualer Schichtvolumenfehler $E_{\text{vol}}(R)$ für verschiedene Tiefenpufferauflösungen für die Bauteile MAT2A, MAT3D sowie das anspruchsvolle Fahrradmodell aus Abbildung 6.5a. Aufgrund der filigranen Verschattungen ist für das Fahrradmodell eine deutlich höhere Auflösung erforderlich.

radmodell eine in der Abbildung rot dargestellte ebene Wand mit einer großen Anzahl an Messpunkten hinzugefügt, welche durch das Fahrrad teilweise verdeckt wird. Von einer einzelnen, mittigen Position der Bahn wird so ein Schatten des gesamten Fahrradmodells auf der dahinterliegenden Wand abgebildet. Durch diese Selbstverschattung wird der Einfluss der Tiefenpufferauflösung direkt am beschichteten Modell in Form von Blockartefakten im Schattenwurf des Fahrrades sichtbar (Abbildung 6.5). Während hier die Unterschiede zwischen den Auflösungen $R = 16^2$, $R = 64^2$ und $R = 1.024^2$ deutlich erkennbar sind, bietet die Auflösung $R = 16.384^2$ nur noch am Bildschirm erkennbare Verbesserungen der Schattendetails. Dieser hohe Selbstverschattungsgrad und die Vielzahl sehr filigraner Strukturen schlagen sich auch bei der Betrachtung der prozentualen Schichtvolumenfehler $E_{h,\text{max}}(R)$ für unterschiedliche Auflösungen R nieder, bei der das Fahrradmodell im Vergleich mit den beiden Matrizenbauteilen durchgehend deutlich höhere Fehlerwerte aufweist (Abbildung 6.6). Derartig feine Strukturen sind jedoch auch in der Praxis kaum beschichtbar. Für die tatsächlich gefertigten Matrizenbauteile liegt der Schichtvolumenfehler bei einer Auflösung von $R = 256^2$ bereits bei unter 0,1% und bei $R = 1.024^2$ unter 0,036%.

Insgesamt kann daher festgestellt werden, dass die Verwendung mittlerer Tiefenpufferauflösungen für die Simulation realer Bauteile, beispielsweise im Rahmen einer Bahnplanung, ausreichend sind und für komplexere Bauteile als die vorgestellten Matrizensegmente noch ausreichend Spielraum besteht. Lediglich für eine Analyse der ungewollt auf den Seitenflächen abgelagerten Materialmenge (Overspray) ist die Verwendung besonders hoher Tiefenpufferauflösungen empfehlenswert.

Alternative zum Tiefenpufferalgorithmus: Raycasting

Speziell für die Analyse des Oversprays könnte statt des Tiefenpufferalgorithmus eine direkte Raycasting-Methode zur Sichtbarkeitsberechnung, welche auch im allgemeinen Fall das Tiefenpufferverfahren ersetzen könnte, vorteilhaft sein. Mittels eines Raycasting-Verfahrens könnte hier eine Entkopplung von der Bildraumauflösung erreicht und jeder Messpunkt durch die Berechnung der Schnittpunkte eines Strahls zwischen Brenner und Messpunkt direkt auf Sichtbarkeit getestet werden.

Nachteilig an der Verwendung eines Raycasting-Verfahrens ist jedoch, dass diese bislang deutlich mehr Rechenzeit als das GPU-seitig implementierte Tiefenpufferverfahren benötigen und daher dem in dieser Arbeit angestrebten Ziel, eine möglichst performante Simulation zu ermöglichen, entgegenstehen. Nichtsdestotrotz gibt es vielversprechende Bestrebungen, hardwarebeschleunigte und echtzeitfähige Raycasting- und Raytracing-Verfahren zu entwickeln [74]. Viel Beachtung findet in diesem Zusammenhang momentan die von Nvidia entwickelte Softwarebibliothek Optix, deren Performanz im Vergleich mit der in dieser Arbeit entwickelten Methodik in Kapitel 6.2.2 diskutiert wird.

Abgesehen von der zur Zeit noch geringeren Performanz von Raycasting-Verfahren besteht jedoch auch ein qualitatives Problem darin, dass viele Schnitttest-Algorithmen für den Dreiecks-Strahl-Test nicht wasserdicht sind [175]. Dies bedeutet, dass die verfolgten Sichtstrahlen auch intakte, geschlossene Dreiecksnetze aufgrund numerischer Probleme hin und wieder durchdringen können, wenn der Schnittpunkt zu nahe an einer Dreieckskante liegt. Diese Problematik trifft auch auf das Raycasting-Modul „Optix Prime“ der Optix-Bibliothek zu. Praktische Tests mit diesem Modul zeigten darüber hinaus, dass eine Verschärfung des Problems im Anwendungsfall der Beschichtungssimulation dadurch entsteht, dass nicht in erster Linie Punkte, die unter besonders flachem Winkel betrachtet werden, falsch klassifiziert werden, sondern beliebige Punkte. Dies kommt zwar selten vor, allerdings ist der dadurch verursachte Beschichtungsfehler meist deutlich größer als bei Fehlklassifizierungen des Tiefenpufferverfahrens, da letztere fast ausschließlich in Bereichen flacher Beschichtungswinkel auftreten, für die das Schichtinkrement modellbedingt ohnehin nahe bei Null liegt.

Wasserdichte Tests für maximale Genauigkeit liefert beispielsweise der von Woop et al. entwickelte Schnitttest [175] oder der von Woop et al. ebenfalls als sehr gut getestete Algorithmus von Dammertz und Keller [36]. Der vergleichsweise hohe Aufwand zur Umsetzung dieser Algorithmen scheint jedoch für die entwickelte Simulation nicht lohnenswert.

6.2 Simulationsgeschwindigkeit

Insbesondere für den Einsatz der Simulation im Rahmen einer automatisierten Bahnplanung ist die Geschwindigkeit der Simulation von zentraler Bedeutung, um die Planung und -optimierung von Beschichtungsbahnen innerhalb eines akzeptablen Zeitfensters zu ermöglichen. Sowohl bei der Entwicklung als auch bei der GPU-basierten Implementierung der entwickelten Algorithmen wurde daher der Geschwindigkeit ein hoher Stellenwert beigemessen.

System / CPU		Grafikhardware	
Hersteller	Intel	Hersteller	Nvidia
CPU	Intel Core i7-2600K	GPU	GeForce GTX Titan black
Architektur	Sandy Bridge	Architektur	Kepler
CPU Kerne	4 (HT 8)	Kerne	2.880
CPU Takt	3,4 GHz	GPU Takt	890 MHz
Hauptspeicher	16 GB DDR3	Speicher	6 GB
		Speichertakt	1.750MHz
Bus	PCIe 2.0	Anschluss	x16
System	MS Windows 7 (x64)	Treiberversion	335.23

Tabelle 6.1: Hardwarekonfiguration des Testsystems.

Während in Kapitel 5.5 bereits an verschiedenen Stellen spezifische Maßnahmen zur Verbesserung der Simulationsgeschwindigkeit diskutiert wurden, wird im Folgenden auf die insgesamt erreichbare Geschwindigkeit für verschiedene Beschichtungssituationen eingegangen, um einen Eindruck von der Praxistauglichkeit des Gesamtsystems zu vermitteln.

6.2.1 Laufzeitmessung und Analyse

Als Testsystem für die Laufzeitmessungen wurde ein PC mit *Intel Core i7-2600 CPU* und *Nvidia GTX Titan black* Grafikkarte mit der in der in Tabelle 6.1 aufgeführten Detailkonfiguration eingesetzt. Die Geschwindigkeit der Simulation wurde programmatisch durch Messung der Simulationsdauer ermittelt. Dabei wurde durch Einfügen des `glFinish`-Befehls am Ende der Simulation sichergestellt (vergleiche Kapitel 5.5.3), dass auch die auf der GPU ausgeführten Algorithmen innerhalb der gemessenen Zeitspanne beendet sind. Die gemessene Zeitspanne umfasst also alle CPU- und GPU-basierten Berechnungen. Ausgenommen sind in der folgenden Betrachtung einmalige Initialisierungen, etwa von Framebuffer Objects oder Shaderkompilierungszeiten, welche in typischen Anwendungsszenarien wie der Bahnplanung oder der Sensitivitätsanalyse aufgrund der Vielzahl von Simulationsaufrufen vernachlässigbar sind. Neben der Dauer einer Simulation ergibt sich als wichtige Kenngröße die Anzahl der durchgeführten Simulationszeitschritte pro Sekunde (SPS) direkt als Quotient der Gesamtanzahl der durchgeführten Schritte und der Simulationsdauer.

Die Laufzeit der Simulation wurde unter Variation der Anzahl der Messpunkte und der Auflösung des verwendeten Tiefenpuffers durchgeführt, um deren Einfluss auf die Laufzeit abzubilden. Zu diesem Zweck wurden kugelförmige Bauteilnetzungen mit 2^k näherungsweise gleichverteilten Messpunkten generiert, wobei k im Bereich 4–21 und dementsprechend die Messpunktanzahl zwischen 16 und 2.097.152 lag. Die Tiefenpufferauflösung wurde zwischen 256^2 und 16.384^2 in den Schritten 256^2 , 1.024^2 , 4.096^2 und 16.384^2 variiert, wobei letztere die höchstmögliche auf der verwendeten GPU unterstützte Auflösung darstellt. Für jede Kombination aus Messpunktanzahl und Tiefenpufferauflösung wurde die Simulation unter Verwendung der in Abbildung 5.9 dargestellten helixförmigen Bahn dreimal mit und dreimal ohne die Berechnung bahnbezogener Zielgrößen in je 25.000 Zeitschritten durchgeführt. Die gemessenen Werte für identische Parametrisierungen wurden für die Auswertung der Laufzeit

ten gemittelt. Die Abbildungen 6.7a und 6.7c zeigen die gemessenen Zeitschritte pro Sekunde sowie die Gesamtlaufzeit der Simulation in Abhängigkeit der in logarithmischer Skala aufgetragenen Vertexanzahl für die Simulationen einschließlich der Berechnung der bahnbezogenen Zielgrößen, wobei die unterschiedlichen Farben verschiedene Tiefenpufferauflösungen kennzeichnen. Die höchsten erreichbaren SPS-Werte liegen dabei im Bereich zwischen 2.700 und 3.700. Eine Analyse mittels der bereits in Kapitel 5.5.3 vorgestellten Software Parallel Nsight des Grafikkartenherstellers Nvidia ergab, dass die Simulation in diesem Bereich, also für Tiefenpufferauflösungen 256^2 und 1.024^2 bis ca. 32.768 Vertices und für die Tiefenpufferauflösung 4.096^2 bis ca. 4.096 Vertices, durch die CPU beschränkt ist. Dies bedeutet, dass die GPU in diesen Fällen die Berechnung der Simulationsergebnisse schneller durchführen kann, als die CPU die OpenGL-Befehle übermittelt. Die leichte Abnahme der SPS-Werte in diesem CPU-beschränkten Bereich ist auf die mit jeder Vervierfachung der Vertexanzahl hinzukommende Reduktionsiteration, welche jeweils 7 zusätzliche API-Befehle beinhaltet, zurückzuführen. Da die GPU deutlich mehr Rechenleistung besitzt und im Rahmen des entwickelten Simulationsalgorithmus nahezu alle Berechnungen durchführt, ist diese CPU-Beschränkung zwar prinzipiell nicht optimal im Bezug auf eine effiziente Nutzung der GPU, allerdings werden in der Praxis kaum Anwendungsfälle in diesen Parameterbereich fallen.

Für höhere Auflösungen des Tiefenpuffers sowie eine größere Anzahl an Messpunkten steigt der GPU-seitige Rechenaufwand und die SPS-Werte sinken deutlich ab. Mit steigender Vertexanzahl fällt die Leistung dabei zunächst für größere Tiefenpuffer schneller, bevor sich die SPS-Werte für die verschiedenen Tiefenpufferwerte schließlich annähern. Für die maximale Tiefenpufferauflösung 16.384^2 werden maximal ca. 500 SPS erreicht und die Gesamtleistung ist stets durch die GPU beschränkt. Dieser deutliche Einfluss der Tiefenpufferauflösung auf die Rechenzeit erscheint insbesondere vor dem Hintergrund, dass die Tiefenpufferauflösung im Gegensatz zur Vertexanzahl nur in einem einzigen Schritt – der Initialisierung des Tiefenpuffers mittels einfachster Shaderprogramme – zum Tragen kommt, zunächst überhöht. Erklärbar ist dieser Einfluss jedoch mit einer Betrachtung der Datenmenge, welche bei der Initialisierung des Tiefenpuffers im Vergleich zu anderen Schritten des Algorithmus in den Grafikspeicher transferiert werden muss. Für die Tiefenpufferauflösungen 256^2 und 1.024^2 sind noch übliche 0,25 bzw. 4 Megabyte, für 4.096^2 bereits 64 MB und für 16.384^2 sogar 1.024 MB pro Zeitschritt zu transferieren. Demgegenüber liegt die berechnete und in den verschiedenen Framebuffern zu speichernde Datenmenge in den übrigen Teilen des Algorithmus im Mittel bei vergleichsweise geringen 187 MB pro einer Millionen Messpunkte für den betrachteten Bereich. Dies umfasst sowohl die bei der Berechnung der oberflächenbezogenen Werte in „Pass 1“ (vergleiche Kapitel 5.5) als auch die bei der Berechnung der bahnbezogenen Werte durch den Reduktionsalgorithmus anfallenden Daten. Insbesondere für den ein Gigabyte großen Tiefenpuffer sind die erreichten 500 SPS daher aus hardwaretechnischer Sichtweise ein sehr guter Wert, da die somit mindestens zwischen GPU und Grafikspeicher transferierten logischen 500 GB/s die theoretisch maximale Speicherbandbreite der GPU von 336 GB/s dank der mit der Fermi-Architektur eingeführten Delta-Color-Compression sogar deutlich übersteigt.

6. EVALUATION

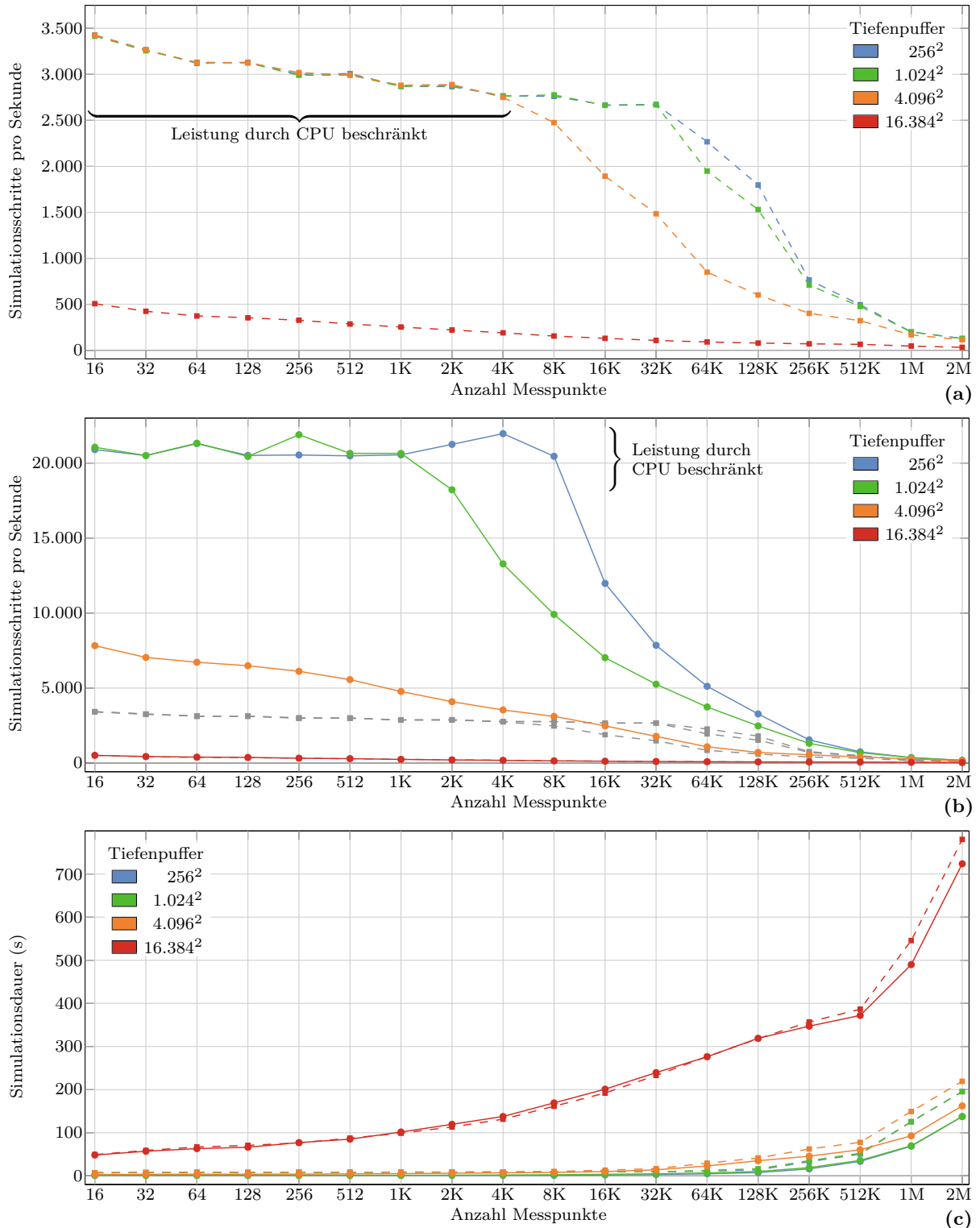


Abbildung 6.7: Erreichte SPS auf dem Testsystem (Tabelle 6.1) in Abhängigkeit von der Tiefenpufferauflösung und der Messpunktanzahl ($K = 1024$, $M = 1024^2$). (a) Erreichte SPS für die Berechnung oberflächen- und bahnbezogener Zielgrößen zusammengenommen (gestrichelte Kurven). (b) Zusätzlich Angabe der erreichten SPS bei Deaktivierung der Berechnung bahnbezogener Zielgrößen (durchgehende Kurven). (c) Gesamtsimulationsdauer für 25.000 Zeitschritte.

Abbildung 6.7b zeigt die erreichten SPS-Werte für dieselben Messpunktanzahlen und Tiefenpuffergrößen wie Abbildung 6.7a, jedoch ohne die Berechnung der bahnbezogenen Zielgrößen. Zusätzlich sind die Kurven aus Abbildung 6.7a hier noch einmal in grau eingezeichnet. Aufgrund der mit 58 API-Aufrufen deutlich geringeren Gesamtzahl an Befehlen werden für die Berechnung der Oberflächengrößen, im wiederum CPU-beschränkten Bereich, 20.000 bis 22.000 SPS erreicht. Der qualitative Verlauf der SPS-Werte ähnelt der in Abbildung 6.7a und die Tiefenpufferauflösung und Vertexanzahl wirken sich entsprechend aus. Auffällig ist lediglich, dass die Tiefenpufferauflösung 4.096^2 in diesem Fall durchgehend GPU-beschränkt ist und auch bei geringeren Auflösungen die SPS-Rate bereits bei deutlich kleineren Vertexanzahlen abnimmt. Dies ist direkt auf die geringere Rechenzeit der CPU für die API-Aufrufe zurückzuführen.

Insgesamt zeigen die gemessenen Laufzeiten eindrucksvoll die Leistungsfähigkeit des GPU-basierten Simulationsansatzes und ermöglichen die Durchführung von Beschichtungssimulationen für relativ komplexe Werkstücke innerhalb weniger Sekunden. Für das in Anhang D dargestellte Bauteil MAT3D, welches 18.231 Messpunkte auf der Bauteiloberfläche aufweist, beträgt die durchschnittliche Simulationsdauer (10.000 Zeitschritte) zur Berechnung der in der in Kapitel 6.4.2 vorgestellten Bahnoptimierung benötigten Schichtdickenverteilung beispielsweise weniger als 0,7s. Auch für komplexere Werkstücke und längere Roboterbahnen ist die benötigte Simulationsdauer ausreichend kurz, um eine Vielzahl von Simulationen im Rahmen einer Bahnoptimierung in kurzer Zeit zu bewältigen.

6.2.2 Vergleich mit Nvidia Optix

Vor dem Hintergrund der rasanten Entwicklung von GPGPU-Verfahren und Bibliotheken wie CUDA¹ oder OpenCL² stellt sich die Frage, ob die Entwicklung einer GPU-beschleunigten Simulation auf Basis der herkömmlichen OpenGL-Renderingpipeline noch sinnvoll beziehungsweise ebenso leistungsfähig ist. Betrachtet man beispielsweise die Entwicklung von CUDA, welche seit 2007 von Nvidia unter großem Ressourceneinsatz betrieben wird und zu einem sehr effizienten und ausgereiften Programmiersystem geführt hat, so ist diese Frage durchaus berechtigt. Wesentliche Gründe, welche für die Verwendung der klassischen OpenGL-Renderingpipeline sprechen, sind jedoch die vorhandene Dualität zwischen OpenGL-Kamera und Spritzbrenner beziehungsweise OpenGL-Sichtpyramide und Spritzstrahlgeometrie sowie ganz allgemein die Tatsache, dass die Beschichtungssimulation ein Geometrieverarbeitungsprozess ist, für den das OpenGL-Framework besser ausgelegt ist als CUDA. Im Kern bietet OpenGL an dieser Stelle den in hochoptimierter Hardware realisierten Tiefenpufferalgorithmus zur Lösung des für die Simulation zentralen Sichtbarkeitsproblems. Für eine nur auf CUDA- oder OpenCL-basierende Simulationsimplementierung müsste diese algorithmische Lücke zunächst geschlossen werden, wobei davon auszugehen ist, dass eine Lösung, welche auch nur näherungsweise die Performanz des hardwaregestützten Tiefenpufferverfahrens er-

¹<http://www.nvidia.de/cuda>

²<http://www.khronos.org/opengl>

reicht, falls überhaupt, dann nur unter sehr hohem Aufwand realisierbar ist. Eine Möglichkeit, diese Lücke zu schließen, bietet die momentan viel beachtete und beworbene Raytracing-Bibliothek „Nvidia Optix“, welche von Nvidia seit einiger Zeit mit dem Ziel, GPU-basiertes Echtzeitraytracing zu realisieren, entwickelt wird und welche seit 2014 die Low-Level-API „Optix Prime“ beinhaltet, um GPU-beschleunigte Sichtbarkeitsberechnungen auf Basis von Schnitttests zwischen Strahlen und Dreiecksnetzen durchzuführen.

Ungeachtet der Umstände, welche sich durch die fehlende Wasserdichtigkeit des von Nvidia eingesetzten Schnitttests ergeben (vergleiche Kapitel 6.1.3), wurde die Leistungsfähigkeit dieser Bibliothek untersucht und mit der der Renderingpipeline-basierten Simulation verglichen. Zu diesem Zweck wurde eine möglichst effiziente Realisierung der Sichtbarkeitsberechnung mittels Optix Prime angestrebt, auf die Reimplementierung des Beschichtungsmodells wurde an dieser Stelle zunächst verzichtet.

Die Optix-Prime Programmierschnittstelle ist sehr einfach gehalten. In einem ersten Schritt wird ein Dreiecksnetz mittels Knotenkoordinaten- und Facetten-Knoten-Inzidenzliste spezifiziert, welches auf die GPU übertragen und mit einer automatisch generierten Begrenzungsvolumenhierarchie zur Durchführung von Schnitttests verknüpft wird. Die parallele Ausführung einer Menge von Schnitttests, welche **query** genannt wird, folgt dann folgendem einfachen Schema:

```
1 Model model = context->createModel();
2 // ... Modell spezifizieren ...
3
4 float* device_rayBuffer, device_hitBuffer;
5 // ... Strahlen definieren ...
6
7 query = model->createQuery(RTP_QUERY_TYPE_ANY);
8 query->setRays( numberOfRays, RTP_BUFFER_FORMAT_RAY_ORIGIN_TMIN_DIRECTION_TMAX,
9               RTP_BUFFER_TYPE_CUDA_LINEAR, device_rayBuffer );
10 query->setHits( numberOfRays, RTP_BUFFER_FORMAT_HIT_T,
11               RTP_BUFFER_TYPE_CUDA_LINEAR, device_hitBuffer);
12 query->execute( RTP_QUERY_HINT_ASYNC );
13
14 // ... HitBuffer auswerten...
```

Dabei werden zunächst die zu testenden Strahlen mit Anfangspunkt, Richtung und zu testenden Intervallgrenzen in einem CUDA-Buffer **device_rayBuffer** abgelegt. Für die Rückgabe der Ergebnisse der Schnitttests, welche im einfachsten Fall boolesche Werte sind, wird ebenfalls ein CUDA-Buffer **device_hitBuffer** angelegt und beide werden an ein für das Modell erzeugtes **query**-Objekt übergeben. Für dieses wird mittels **RTP_QUERY_TYPE_ANY** die schnellste Schnitttestvariante gewählt, welche einen beliebigen Schnitt sucht. Mittels **execute** wird der Test schließlich ausgeführt und anschließend können die Ergebnisse anhand des Hitbuffers ausgewertet werden. Für die Bestimmung der zu beschichtenden Messpunkte werden dabei Strahlen von allen Messpunkten in Richtung der Brennerposition in Intervallen ausgewertet, welche minimal oberhalb der Werkstückoberfläche beginnen und bis zur Brennerposition

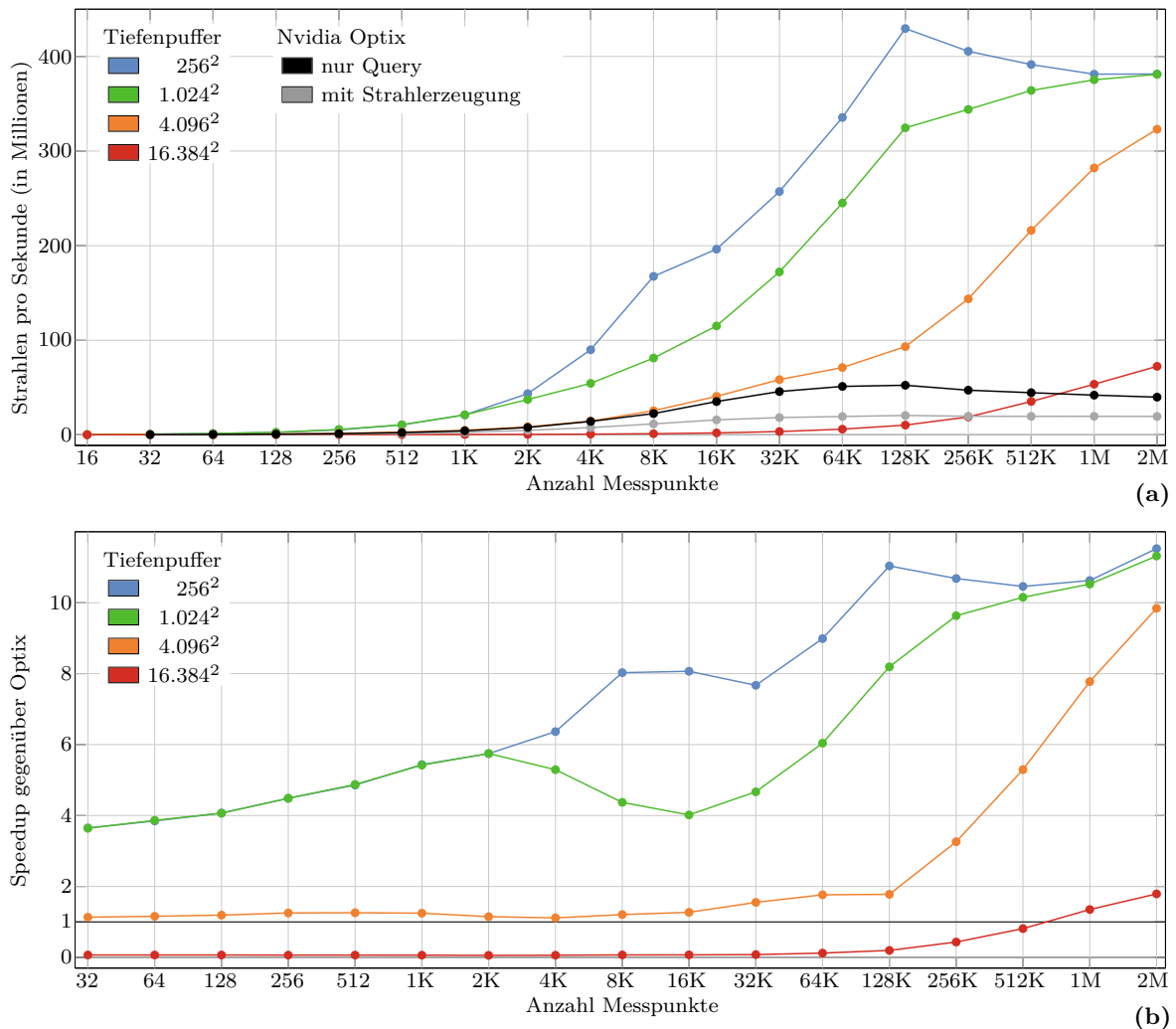


Abbildung 6.8: Vergleich der Geschwindigkeit der Beschichtungssimulation mit Raycasting-basierter Sichtbarkeitsberechnung durch Nvidia Optix Prime. (a) Erreichte Strahltests pro Sekunde für verschiedene Tiefenpufferauflösungen der Simulation sowie für Optix Prime mit und ohne Neugenerierung von Strahlen in jedem Zeitschritt. (b) Speedup-Faktor der Tiefenpuffer-basierten Sichtbarkeitsberechnung gegenüber Optix Prime ohne Strahlneugenerierung.

reichen. Hierfür müssen die Startpunkte, Richtungen und Intervallgrenzen jedes Strahls in jedem Zeitschritt neu berechnet und gesetzt werden. Diese Berechnungen erfolgten CPU-seitig unter Ausnutzung aller CPU-Kerne und zur Maximierung der CPU-GPU Transferrate in „page-locked“-Speicher, welcher stets in physikalischem Arbeitsspeicher verfügbar gehalten wird. Eine umgekehrte Strategie, bei der das Modell relativ zum Brenner transformiert und stets dieselben Strahlen verwendet werden, um das Tiefenpufferverfahren nachzubilden, ist nicht praktikabel, da in diesem Fall in jedem Zeitschritt die Begrenzungsvolumenhierarchie neu aufgebaut werden müsste.

Für die Messung der Laufzeiten dieser Methode wurden die bereits bekannten Helixbahnen und Kugelbauteile mit unterschiedlich vielen Messpunkten verwendet (Abbildung 5.9). Um zu

berücksichtigen, dass die Auswertung der Schnitttests und die Auswertung des Simulationsmodells auch GPU-seitig in CUDA erfolgen könnte, wobei keine teuren Speichertransfers anfallen würden, wurden die Laufzeiten zweier Varianten – mit und ohne Neuberechnung der Strahlen – untersucht. Für die Variante ohne eine Strahlneuberechnung wurden stets dieselben, GPU-seitig gespeicherten Strahlen verwendet. Die auf dem Testsystem gemessenen Geschwindigkeiten sind in Abbildung 6.8a im Vergleich mit den zuvor ermittelten Geschwindigkeiten der Beschichtungssimulation dargestellt. Entsprechend üblicher Raytracing-Benchmarks erfolgt die Angabe hier in Millionen Strahlen pro Sekunde (Mrays/s). Die Optix-basierten Schnitttests erreichen maximale 52,4 Mrays/s ohne Speichertransfers und lediglich 20,5 Mrays/s und bleiben damit deutlich hinter den Marketingwerten für Optix zurück³. Demgegenüber erreicht die implementierte, auf der Rendering-Pipeline basierende Simulation weitaus bessere Geschwindigkeiten mit einem Maximum von 429 Mrays/s, wobei hier zusätzlich bereits die Berechnung und das Speichern sämtlicher oberflächenbezogener Zielgrößen enthalten sind. Für die Tiefenpufferauflösungen $R \leq 4.096^2$ ist die Simulation in keinem Fall langsamer als Optix Prime, für $R \leq 1.024^2$ ist die Simulation stets mindestens $3\times$ und für Bauteile mit sehr vielen Messpunkten bis zu $9.5\times$ schneller als die Optix-Implementierung. Wird in der Shader-Implementierung der Beschichtungssimulation die Berechnung der Zielgrößen deaktiviert, um auch hier lediglich die Sichtbarkeit der Messpunkte zu berechnen und zu speichern, werden sogar bis zu 578 Mrays/s erreicht. Der in diesem Fall erreichte Speedup-Faktor gegenüber der schnelleren Optix-Variante ist in Abbildung 6.8b dargestellt und liegt im relevanten Bereich ab etwa 16.384 Messpunkten zwischen $4\times$ und $11,5\times$ mit einem Durchschnitt von $10,25\times$. Die Verwendung der herkömmlichen OpenGL-Renderingpipeline zur Abbildung der Funktionalität der Beschichtungssimulation und die damit verbundene Ausnutzung des hardwaregestützten Tiefenpufferverfahrens ist folglich um ein Vielfaches schneller als die Verwendung von Optix Prime, welches seinerseits als aktueller Stand der Technik auf dem Gebiet des GPU-basierten Raytracings angesehen werden kann.

6.3 Beschichtung von Tiefziehwerkzeugen

Zur Evaluation der entwickelten Beschichtungssimulation anhand komplexer Bauteilgeometrien wurden die in Anhang D dargestellten Matrizenbauteile MAT2A und MAT3D eines im Rahmen des Sonderforschungsbereichs 708 entwickelten Tiefziehwerkzeugs verwendet.

Im folgenden Abschnitt 6.3.1 wird dazu zunächst auf den Versuchsaufbau und die Versuchsdurchführung eingegangen. Abschnitt 6.3.2 behandelt die Vermessung der Schichtdickenverteilung auf den Bauteilen, welche aufgrund der hohen geometrischen Komplexität und der vergleichsweise dünnen Beschichtung eine nichttriviale Aufgabe darstellt. Anschließend wer-

³Weitere Tests zeigten, dass die seitens Nvidia angegebenen mehreren hundert Mrays/s lediglich in sehr einfachen Spezialfällen erreichbar sind, beispielsweise für gleichlange parallele Strahlen oder vielfache Tests mit demselben Strahl. Es ist zu vermuten, dass die unvermeidbar ungleiche Berechnungsdauer für unterschiedliche Strahlen die parallelisierte Berechnung stark ausbremst.

den in Abschnitt 6.3.3 die Messergebnisse mit den Simulationsergebnissen verglichen und diskutiert.

6.3.1 Bauteile und Versuchsdurchführung

Zur Evaluation der Simulation wurden zwei unterschiedliche Matrizensegmente beschichtet und vermessen, welche mit den Kürzeln MAT2A und MAT3D bezeichnet werden. Beide Bauteile sind in Anhang D ausführlich in mehreren Darstellungen illustriert, wobei die einzelnen Abbildungen jeweils die Bauteilbemaßung, die Einordnung des Segments innerhalb der vollständigen vierelementigen Matrize des Werkzeugs, die Definition der zu beschichtenden Funktionsfläche (Region of Interest) sowie mehrere Ansichten der zur Beschichtung verwendeten Roboterbahn zeigen. Letztere umfassen für jedes Bauteil orthographische Seiten- und Vorderansichten der Roboterbahn sowie zwei perspektivische Ansichten, welche zum einen den Bahnverlauf an der Brennerdüse, im Abstand von 110 mm von der Bauteiloberfläche, und zum anderen den Bahnverlauf des TCP auf der Bauteiloberfläche darstellen.

Erstellt wurden beide Bahnen zunächst mittels des von Kout und Müller [82] beschriebenen Algorithmus zur Berechnung quasi-paralleler Bahnen auf Basis einer Dreiecksnetzrepräsentation. Als Ausgangskurven (Seed-Kurven) wurden hierfür jeweils die Bauteilkanten am unteren Ende der Funktionsfläche unterhalb des Ziehradius gewählt und der geodätische Bahnabstand auf den für den Lichtbogenspritzprozess gewünschten Wert von 5 mm festgelegt. Abweichend von dem üblichen Ziel, die gesamte Funktionsfläche der Bauteile mit einer konstanten Schichtdicke zu beschichten, wurde für die in dieser Arbeit durchgeführten Evaluationsversuche die Bahn des Bauteils MAT2A gekürzt, um ein sichtbares Auslaufen der Beschichtung auf der Oberseite des Bauteils zu realisieren (Seite 245 unten). Dadurch wird ein weiteres Beschichtungsmerkmal eingeführt, welches beispielsweise im Rahmen der Vermessung der Schichtprofile als Bezugspunkt dienen kann.

Im Rahmen der praktischen Durchführung der Beschichtungsexperimente wurden drei Proben vom Typ MAT2A und zwei Proben vom Typ MAT3D mittels des in Kapitel 2.2 beschriebenen Lichtbogenspritzprozesses beschichtet. Wie auch bei der Herstellung von Footprintprofilen wurden die Proben zur Vorbereitung zunächst händisch gesandstrahlt, um die Oberfläche anzurauen. Da damit auch ein geringer Schichtabtrag des Grundwerkstoffs einhergeht, wurden die Proben nach dem Sandstrahlen mit dem im folgenden Kapitel beschriebenen, taktilen Scansystem vermessen, um eine exakte digitale Referenzgeometrie jeder unbeschichteten Probe für die spätere Bestimmung der Schichtdicken zu erhalten. Um zu verhindern, dass einzelne Partikel an den nicht zu beschichtenden Seitenflächen anhaften, wurden diese Flächen anschließend mit einem speziellen Maskierungsmittel bestrichen. Im finalen Produktionsprozess ist dies erforderlich, damit die einzelnen Matrizensegmente formschlüssig montiert werden können. Für die hier durchgeführte Evaluation ist dieser Maskierungsschritt primär darin begründet, dass die Seitenflächen des Bauteils als geometrische Referenzflächen für die Vermessung der Bauteile dienen. Zur Sicherstellung einer exakt identischen Einmessung der Bauteile im taktilen Scanner vor und nach dem Beschichtungsprozess müssen diese Flächen

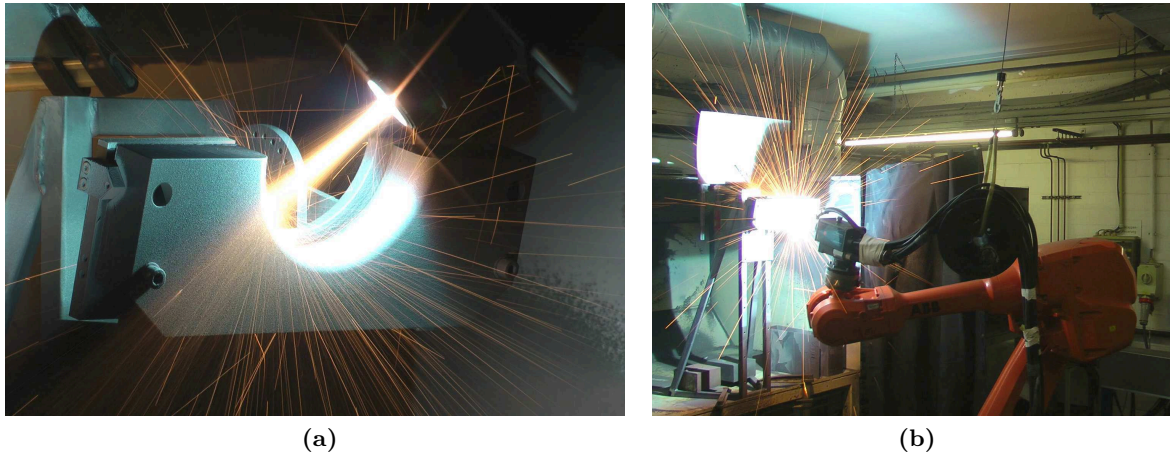


Abbildung 6.9: Beschichtung des Bauteils MAT3D mit dem Lichtbogensspritzprozess. (a) Detailansicht der mit nach oben weisendem Radiusausschnitt in der Halterung montierten Probe im Beschichtungsprozess. (b) Stellung des Roboters mit Führung des Schlauchpakets über eine mit einem Federzug an der Decke aufgehängte Rolle.

daher unverändert bleiben. Nach der Maskierung der Seitenflächen wurden die Proben in einem Ofen vortemperiert und anschließend beschichtet (Abbildung 6.9). Um eine konsistente Platzierung der Proben relativ zum Roboter zu gewährleisten, wurden alle Proben während der Beschichtung in Magnethalterungen mit definierten seitlichen Anschlägen platziert. Für eine möglichst exakte Nachstellung des Beschichtungsprozesses in der Beschichtungssimulation wurde für die Roboterbahnen beider Bauteile zunächst die Roboterdynamik mit der Software Robotstudio des Roboterherstellers simuliert. Dabei entsprachen die relative Platzierung von Roboter und Bauteil sowie die Posen des Roboters exakt denen der Versuchsdurchführung. Die resultierenden, dynamikkorrigierten Bahnen wurden anschließend in die Beschichtungssimulation überführt. Die Längen der resultierenden TCP-Bahnen für die Bauteile MAT2A und MAT3D betragen für jeweils zwei durchgeführte Überläufe 13,5 m respektive 37,9 m und sind in 3.020 bzw. 7.815 diskreten Roboterkonfigurationen kodiert. Die Verfahrzeiten betragen 71,6 s respektive 183,6 s. Für die Bauteilgeometrien wurden Dreiecksnetze auf Basis der CAD-Daten der Bauteile erstellt, welche 15.863 (MAT2A) und 18.231 (MAT3D) Vertices beziehungsweise Messpunkte enthalten.

6.3.2 Messung der Schichtdickenverteilung

Wie bereits in Kapitel 3 deutlich wurde, ist die großflächige Vermessung der Schichtdickenverteilung bereits für ebene Footprintssubstrate ein komplizierter Prozess, welcher einer Vielzahl von potentiellen Störeinflüssen unterliegt. Hierzu zählen die spannungsbedingte Bauteil deformation, der Schichtabtrag und die Aufrauung durch das Sandstrahlen, Positionierungs- und Einmessfehler sowie im Allgemeinen das Problem, eine im Verhältnis zur Größe der Bauteiloberfläche sehr dünne Schicht zu messen. Bei der Vermessung komplexer Geometrien kommt darüber hinaus das Problem hinzu, dass das Erschließen einer Nullhöhe auf Basis

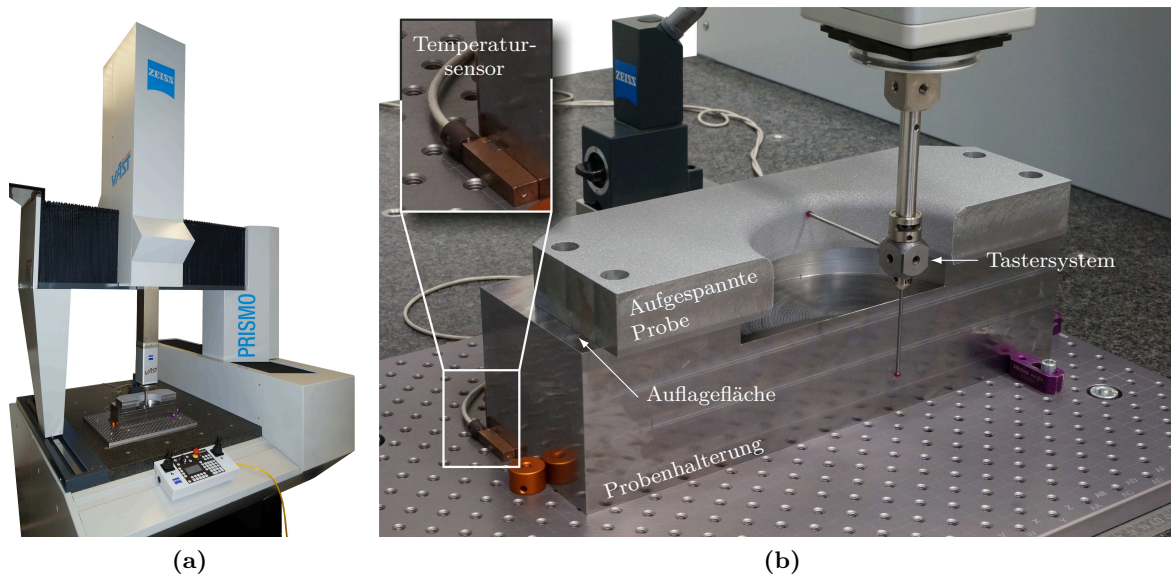


Abbildung 6.10: 3D-Schichtdickenvermessung für das Bauteil MAT3D. (a) Koordinatenmessmaschine Zeiss Prismo mit MAT3D-Werkstück. (b) Aufgespanntes Werkstück in der Messumgebung. Die seitlich überstehende Auflagefläche dient der vertikalen Probeneinmessung. Temperatursensoren sind an der Halterung und am Werkstück befestigt und der Scanvorgang erfolgt mit einem 2-Taster-System (vertikal/horizontal), um alle Messpunkte zu erreichen.

eines unbeschichteten Bauteilrandes nicht möglich ist. Eine Vermessung der äußeren Form des beschichteten Bauteils und die Berechnung der Schichtdickenverteilung als Differenz zur Form der CAD-Geometrie des Bauteils ist theoretisch möglich, berücksichtigt allerdings weder den lokal oder von Probe zu Probe variierenden Abtrag durch das händische Sandstrahlen noch Fertigungstoleranzen in der Bauteilfertigung oder eine Verformung durch freiwerdende Eigenspannungen. Um diese Einflussgrößen nicht zu vernachlässigen, wurden alle im Rahmen dieser Evaluation beschichteten Probengeometrien je einmal vor der Beschichtung, aber nach dem Sandstrahlprozess, und einmal nach der Beschichtung vermessen und die Schichtdicken als Differenzwerte berechnet.

Taktile 3D-Schichtdickenvermessung

Als Scanner kam dabei ein Portalscanner vom Typ „Prismo 7“ mit taktilem Messkopf „Vast Gold“ der Firma Zeiss⁴ zum Einsatz (Abbildung 6.10). Diese hochgenaue Koordinatenmessmaschine weist eine maximale Antastabweichung von $1,4\ \mu\text{m}$ sowie eine Längenmessabweichung von $\text{MPE}(\text{E}0) = 1,4 + L/333\ \mu\text{m}$ (nach DIN EN ISO 10360-2) auf, wobei L die zu messende Länge in mm angibt. Um temperaturbedingten Messabweichungen vorzubeugen, ist der Scanner in einer vollklimatisierten Messkabine aufgestellt und verfügt zudem über Temperatursensoren zur Überwachung der Proben temperatur während des Scanvorgangs. Die gescannten MAT2A- und MAT3D-Proben wurden daher vor jedem Scanvorgang mindestens

⁴<http://www.zeiss.de>

zwölf Stunden akklimatisiert. Die zu erwartende Genauigkeit dieses Scanners übertrifft die der in Tabelle 3.1 in Kapitel 3.5.1 aufgeführten Scanner für die Vermessung vergleichsweise großer Bauteile.

Mittels der umfangreichen Scansoftware „Zeiss Calypso“ wurde für jede der beiden Geometrien ein Scanprogramm entwickelt, welches im ersten Schritt die Lage der Probe im Scanbereich durch Antasten nicht beschichteter Seitenflächen der Probe ermittelt. Dabei werden die Seitenflächen und die dem Ziehradius gegenüberliegende Rückseite der Probe jeweils mehrfach angetastet, um ein Probenkoordinatensystem zu definieren. Da die gesamte Oberseite der Proben beschichtet ist, wurde zur vertikalen Einmessung die seitlich über die Probe hinausragende Auflagefläche der eigens konstruierten und in Abbildung 6.10b für MAT3D dargestellten Probenhalterung verwendet. Diese besonders massiv ausgelegten Halterungen, mit denen die Bauteile unterseitig fest verschraubt werden, erfüllen darüber hinaus den Zweck geringfügigen, thermisch bedingten Verformungen, welche aus dem Beschichtungsprozess hervorgehen, entgegenzuwirken.

Zur zusätzlichen Kontrolle eventuell auftretender Verformungen sowie der vertikalen Probenlage wurden Scanbahnen auf der wenige Millimeter über die Aufspannung hinausragenden Unterseite der Proben sowie auf dem unbeschichteten Teil der Oberseite der MAT2A-Bauteile programmiert. Die Auswertung dieser Bahncans zeigte eine geringfügige Durchbiegung der Proben, welche jedoch vor und nach der Beschichtung identisch war. Zusätzlich wurden vertikale Verschiebungen der Probenkoordinatensysteme im Bereich von 3,25–4,7 μm festgestellt, welche nachträglich korrigiert wurden. Eine mögliche Ursache für diese geringe Abweichung könnten Staubkörner oder Einzelpartikelanhaftungen zwischen der Probenunterseite und der Aufspanfläche der Halterung sein.

Nach der Einmessung des Probenkoordinatensystems folgt im zweiten Schritt des entwickelten Scanprogramms die eigentliche Vermessung der Probenoberfläche. Um hierbei einen einfachen Vergleich von Mess- und Simulationsdaten zu ermöglichen, wurde eine Teilmenge der Vertices des zur Simulation verwendeten Dreiecksnetzes als Antastpunkte für die Koordinatenmessmaschine definiert. Die Antastung erfolgte als Einzelantastung jedes Punktes in Normalenrichtung, welche genauer als eine scannende Abtastung der Oberfläche ist und das wiederholte Antasten desselben Punktes vor und nach der Beschichtung garantiert. Zur Beschränkung der Scandauer einer Probe, welche im Bereich von einigen Stunden liegt, wurden für MAT2A 1.033 der 3.322 und für MAT3D 2.160 der 4.571 Messpunkte innerhalb der Region of Interest gescannt. Im Anschluss wurden die Schichtdickenwerte der nicht gescannten Messpunkte auf Basis der Werte benachbarter Punkte interpoliert, wobei durch die Auswahl der gescannten Messpunkte stets eine ausreichende lokale Abdeckung mit Messwerten sichergestellt war.

Die auf Basis von drei Proben für MAT2A beziehungsweise zwei Proben für MAT3D ermittelten durchschnittlichen, d.h. an jedem Messpunkt gemittelten, Schichtdickenverteilungen sind in den Abbildungen 6.14a und 6.14b dargestellt. Im Folgenden werden diese gemessenen Schichtdickenverteilungen in Bezug auf den taktilen Scanvorgang mit $\mathcal{T}_2(i)$ respektive $\mathcal{T}_3(i)$ bezeichnet, wobei i einen oder ggf. eine Menge von Messpunktindizes angibt. Die Diskussion dieser Schichtdickenverteilungen erfolgt in Kapitel 6.3.3.

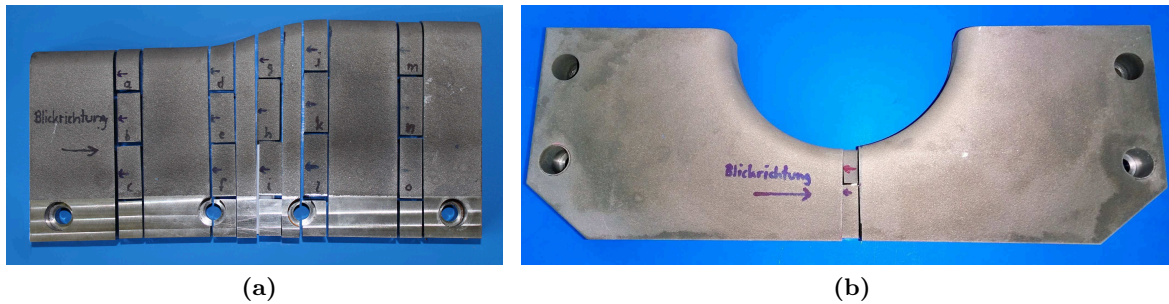


Abbildung 6.11: Positionen der Querschliffe für die Bauteile MAT2A und MAT3D. (a) Fünf vertikal verlaufende Querschliffe – aufgrund der maximalen Einbettungsgröße in je drei Abschnitte unterteilt. (b) Einzelner vertikaler Schnitt durch MAT3D.

Vergleich mit Querschliffmessungen

Zur Absicherung der dreidimensionalen taktilen Scans wurden zusätzlich lichtmikroskopische Querschliffe angefertigt und mit dem bereits in Kapitel 3.7 beschriebenen Softwaretool analysiert, um hochauflösende Profilverläufe zu bestimmen. Aufgrund des mit diesen Messungen verbundenen extrem hohen Zeitaufwandes wurde im Rahmen dieser Querschliffmessungen für jedes Bauteil nur die Probe verwendet, welche am genauesten der mittleren Schichtdickenverteilung aller Proben für dieses Bauteil entspricht. Wie in Abbildung 6.11 dargestellt, wurde die MAT2A-Probe entlang von fünf Schnitten und die MAT3D-Probe entlang eines Schnitts getrennt, wobei jeder Schnitt noch einmal entsprechend der maximal mit diesem Verfahren vermessbaren Probengröße in drei Abschnitte zerteilt wurde. Zusätzlich zu der im vorherigen Abschnitt beschriebenen Einzelpunktvermessung wurden mit der taktilen Koordinatenmessmaschine im Vorhinein auch Höhenprofile entlang genau dieser Querschliff lagen erfasst.

Zur Beurteilung der Abweichungen sowohl zwischen Querschliffprofilen und taktilen Scan als auch zwischen den beiden Scans und den simulierten Werten, welche im folgenden Kapitel erfolgt, ist es sinnvoll, die beschichteten Bauteilflächen in Bereiche mit ähnlichen Eingriffsbedingungen zu segmentieren. Anhand der aus den Scandaten erhaltenen Information sowie der Geometrie der Bauteile wurden die in Abbildung 6.12 dargestellten Segmente gewählt. Zu diesem Zweck wurden die Bereiche „Oberseite“, „Radius“, „Front“ und „Auslaufzone“ – im Folgenden durch die Indexmengen $\mathcal{P}_O, \mathcal{P}_R, \mathcal{P}_F$ und \mathcal{P}_A gekennzeichnet – definiert, wobei das Bauteil MAT2A keinen Front- und Auslaufbereich besitzt. Die Messpunkte der beiden Bauteilgeometrien wurden diesen Bereichen zugeordnet, wobei Messpunkte in erkennbaren Übergangszonen der Schichtdickenverteilung oder der Eingriffsbedingungen nicht zugeteilt wurden, um innerhalb jeder Zone möglichst homogene Bedingungen abzubilden. Des Weiteren wurden alle Messpunkte auf den Kanten des Bauteils nicht zugeordnet, da an diesen keine wohldefinierten Eingriffsbedingungen vorliegen.

Für jedes der Segmente wurde zunächst ein mittlerer Schichtdickenwert sowohl anhand der taktilen Messungen als auch auf Basis der in dem jeweiligen Segment liegenden Querschliffprofile ermittelt, welche in Tabelle 6.2 dargestellt sind. Darin bezeichnen $\mathcal{T}_{2|3}(\mathcal{P}_*)$ die Menge der

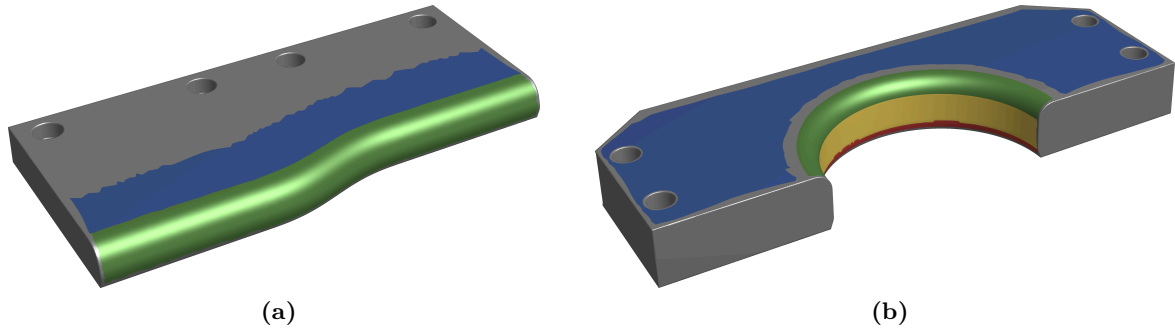


Abbildung 6.12: Segmentierung der Region of Interest der Bauteile in unterschiedliche Bereiche mit ähnlichen Eingriffsparameter. (a) Segmentierung des Bauteils MAT2A in die Bereiche \mathcal{P}_O (Oberseite, blau) und \mathcal{P}_R (Radius, grün). (b) Segmentierung des Bauteils MAT3D in die Bereiche \mathcal{P}_O (Oberseite, blau), \mathcal{P}_R (Radius, grün), \mathcal{P}_F (Front, gelb) und den in rot dargestellten kleinen Auslaufbereich \mathcal{P}_A .

taktil gemessenen Schichtdickenwerte der Messpunkte in \mathcal{P}_* und $\mathcal{Q}_{2|3}(\mathcal{P}_*)$ die Menge aller (ungeordneten) Querschliffmesswerte für die Schichtdicke in dem Bereich – jeweils für das Bauteil MAT2A oder MAT3D. Die Differenzwerte zeigen, dass die taktil erfassten mittleren Schichtdicken der einzelnen Bereiche, mit Ausnahme des Radiusbereichs von MAT3D, im Durchschnitt um $6,0\ \mu\text{m}$ von den Querschliffmessungen abweichen, wobei das Vorzeichen der Abweichung unterschiedlich ist. Die hohe Abweichung von $\mathcal{T}_3(\mathcal{P}_R)$ kann dabei nicht allein auf die Tatsache zurückgeführt werden, dass dieses Segment von allen Segmenten die am wenigsten homogenen Eingriffsbedingungen aufweist, welche potentiell durch einen einzelnen Querschliff nicht ausreichend genau erfasst werden. Stattdessen zeigt sich auch im Vergleich des hochauflösenden taktilen Linienscans entlang der Lage des später angefertigten Querschliffs mit dem Querschliffprofil eine Unterschätzung der Schichtdicke durch den taktilen Scan, welche in anderen Bereichen nicht in dieser Ausprägung auftritt (Abbildung 6.13).

Bauteil	Bereich \mathcal{P}_*	Taktiler Scan $\bar{\mathcal{T}}_{2 3}(\mathcal{P}_*)$	Querschliff $\bar{\mathcal{Q}}_{2 3}(\mathcal{P}_*)$	Differenz $\bar{\mathcal{T}}_{2 3}(\mathcal{P}_*) - \bar{\mathcal{Q}}_{2 3}(\mathcal{P}_*)$	Messwertstreuung $\sigma(\mathcal{T}(\mathcal{P}_*))$
MAT2A	\mathcal{P}_O	346,9	353,7	-6,8	20,0
MAT2A	\mathcal{P}_R	276,1	284,5	-8,4	12,3
MAT3D	\mathcal{P}_O	350,7	348,2	2,5	22,7
MAT3D	\mathcal{P}_R	256,8	280,4	-23,6	38,2
MAT3D	\mathcal{P}_F	364,4	358,6	5,8	53,6
MAT3D	\mathcal{P}_A	301,8	308,3	-6,5	36,0

Alle Angaben in μm

Tabelle 6.2: Mittelwerte $\bar{\mathcal{T}}_{2|3}(\mathcal{P}_*)$ und $\bar{\mathcal{Q}}_{2|3}(\mathcal{P}_*)$ der Schichtdicke in den unterschiedlichen Bereichen \mathcal{P}_* der Bauteile MAT2A und MAT3D gemessen mit der taktilen Koordinatenmessmaschine und als Querschliffprofil. Abgesehen von einem Ausreißer für den Radiusbereich von MAT3D beträgt die durchschnittliche Abweichung $6,0\ \mu\text{m}$. Die letzte Spalte gibt die Streuung der Messwerte $\mathcal{T}(\mathcal{P}_*)$ innerhalb jedes Segments an, welche auf eine Kombination aus Schichtdickenschwankung und rauheitsbedingter Messabweichung zurückzuführen ist.

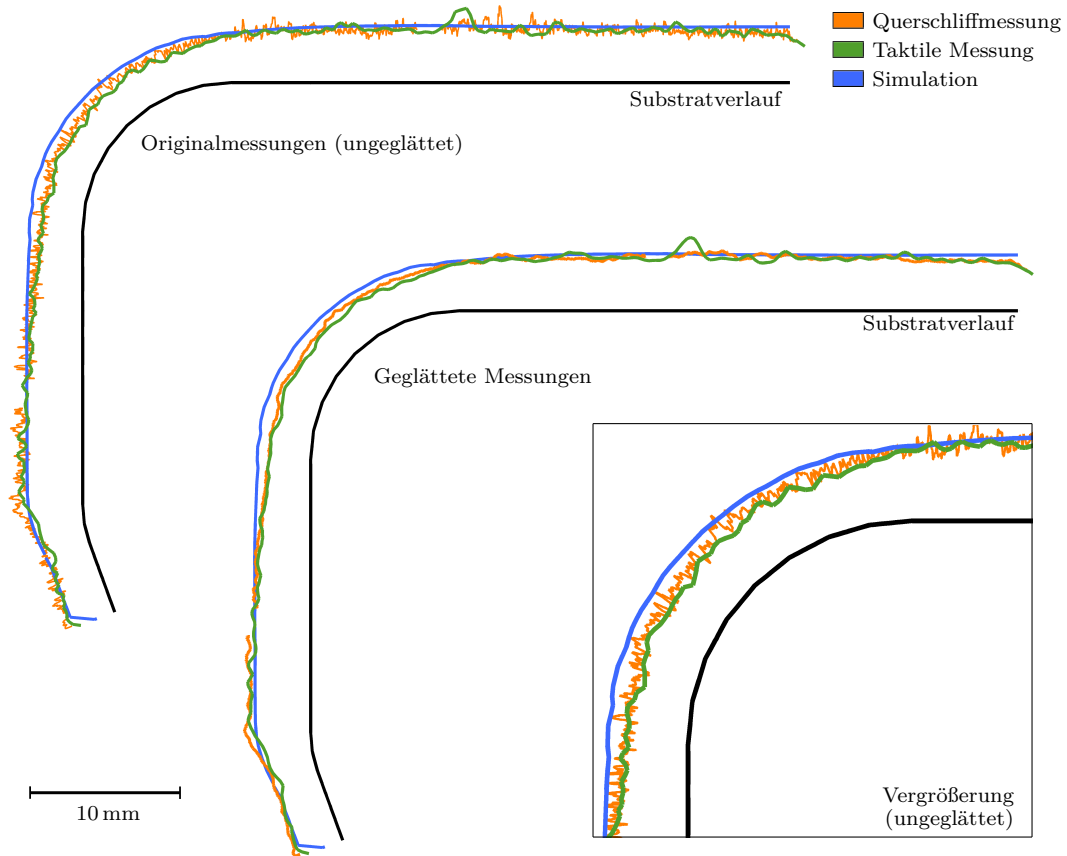


Abbildung 6.13: Links oben: 10-fach überhöht dargestellte Verläufe der im Querschleif und taktile gemessenen Schichtdicken entlang des in Abbildung 6.11b gezeigten Schnitts durch das Bauteil MAT3D. Im Radiusbereich ist die Überschätzung der Schichtdicke durch die Simulation sowie die leichte Unterschätzung durch die taktile Messung zu erkennen. Unten rechts: Vergrößerung des Radiusbereichs mit ungeglätteten Messungen und Simulationsergebnis.

Eine mögliche Ursache für die Abweichungen des taktile Scanvorgangs von den Querschleifprofilen, welche in diesem Fall als Ground-Truth-Basis angesehen werden können, stellen die unterschiedlichen Rauheiten von gestrahlter Substratoberfläche und Beschichtungsoberfläche dar. Da der Scantaster des taktile Scanners, welcher in diesem Fall einen Durchmesser von 4 mm aufwies, auf den Rauheitsspitzen entlangläuft, steigt der Scanfehler mit zunehmender Rauheit der Oberfläche. Für die Schichtdickenmessung nach dem Differenzprinzip bedeutet dies, dass die Schichtdicke durch den taktile Scanvorgang unterschätzt wird, wenn die Substratrauheit höher als die Rauheit der Beschichtung ist und umgekehrt. Untermauert wird diese These dadurch, dass die Rauheit auf dem beschichteten Radius der MAT3D-Probe im Vergleich mit anderen Bereichen gering ausfällt, was die in Tabelle 6.2 aufgeführte hohe Abweichung von $23,6\ \mu\text{m}$ erklären würde. Die geringere Rauheit bzw. das Fehlen größerer

Bauteil	Bereich \mathcal{P}_*	Rauheit R_a	\varnothing Intervallbreite $\omega_{2 3}(\mathcal{P}_*)$
MAT2A	\mathcal{P}_O	23,9	27,2
MAT2A	\mathcal{P}_R	26,4	23,4
MAT3D	\mathcal{P}_O	24,6	22,2
MAT3D	\mathcal{P}_R	20,0	16,1
MAT3D	\mathcal{P}_F	23,6	20,3
MAT3D	\mathcal{P}_A	21,6	17,5

Alle Angaben in μm

Tabelle 6.3: Anhand von Querschliffprofilen ermittelte Rauheit R_a sowie die mittlere Intervallbreite $\omega_{2|3}(\mathcal{P}_*)$ innerhalb jedes Segments \mathcal{P}_* . Die Rauheit R_a kann dabei, aufgrund des sehr genauen Messverfahrens, als frei von Scanfehlern und somit als tatsächliche Schichtrauheit angesehen werden. Die Intervallbreite gibt die Schwankung der Schichtdicke an jedem Einzelmesspunkt über verschiedene Proben desselben Bauteiltyps an.

Rauheitsspitzen ist für den Radius bereits in Abbildung 6.13 erkennbar. Zusätzlich zeigt das in Tabelle 6.3 angegebene und auf Basis der Querschliffe berechnete Rauheitsmaß R_a , welches den durchschnittlichen Abstand der Messpunkte zur Mittellinie wiedergibt, dass auf dem Radius des Bauteils MAT3D eine vergleichsweise geringe Rauheit vorliegt. Zusätzlich ist in der Tabelle die durchschnittlichen Intervallbreite

$$\omega_{2|3}(\mathcal{P}_*) = \frac{1}{|\mathcal{P}_*|} \sum_{i \in \mathcal{P}_*} \left| \left(\max_{\text{Proben}} \mathcal{T}_{2|3}(i) \right) - \left(\min_{\text{Proben}} \mathcal{T}_{2|3}(i) \right) \right|$$

der taktilen Schichtdickenmessung über alle vermessenen Proben für die unterschiedlichen Bereiche \mathcal{P}_* angegeben, welche ebenfalls als indirekter Rauheitsindikator dienen kann.

Unter der Annahme, dass die Abweichungen der taktilen Messungen tatsächlich auf die Oberflächenrauheiten zurückführbar und innerhalb eines Bereichs \mathcal{P}_* eines Bauteils, aufgrund der ähnlichen Eingriffsbedingungen, konstant sind, können die taktilen Messungen durch ein entsprechendes Offset gemäß Tabelle 6.2 korrigiert werden. Für $\mathcal{T}_2(\mathcal{P}_R)$ ergibt sich beispielsweise die korrigierte Schichtdickenverteilung

$$\mathcal{C}_2(\mathcal{P}_R) = \mathcal{T}_2(\mathcal{P}_R) + 8,4 \mu\text{m}.$$

Diese korrigierten Schichtdickenverteilungen $\mathcal{C}_{2|3}(i)$ werden im folgenden Abschnitt zusätzlich zu den unkorrigierten $\mathcal{T}_{2|3}(i)$ als Vergleichswerte für die Simulationsergebnisse verwendet.

Da die Abweichungen der taktilen Vermessung gegenüber den Querschliffmessungen insgesamt in der Größenordnung der Oberflächenrauheiten von sandgestrahlter und beschichteter Oberfläche liegen, ist davon auszugehen, dass für eine weitere Steigerung der Genauigkeit taktiler Schichtmessungen auf komplexen Geometrien eine genauere Untersuchung dieser Rauheitsinflüsse zur Entwicklung einer geeigneten Kompensationsstrategie notwendig ist. Für die im Rahmen dieser Arbeit durchgeführte Schichtdickenvermessung stehen zwar Querschliffprofile zur Verfügung, welche eine Fehlerabschätzung und -korrektur der taktilen Messungen erlauben, grundsätzlich ist die hierfür notwendige Zerteilung komplexer Bauteile jedoch wenig

praktikabel. Für kleinere bis mittelgroße Bauteile kommt hier als alternative Messmethode beispielsweise die in Kapitel 3.6.2 erläuterte Konfokalprofilometrie in Kombination mit einer Abtastsimulation in Frage. Insgesamt kann an dieser Stelle jedoch festgehalten werden, dass die erreichte Genauigkeit der taktilen Messungen im Hinblick auf die erzielte großflächige Oberflächenabdeckung und unter Berücksichtigung der Störeinflüsse gut ist.

6.3.3 Simulationsergebnisse

In diesem Kapitel werden die mit der entwickelten Beschichtungssimulation für die Bauteile MAT2A und MAT3D erzielten Ergebnisse vorgestellt und diskutiert. Im Fokus steht dabei im Folgenden zunächst der Vergleich der simulierten und gemessenen Schichtdickenverteilungen. Im Anschluss werden die sekundären oberflächenbezogenen sowie die bahnbezogenen Simulationsergebnisse vorgestellt. Für alle durchgeführten Simulationen kamen die in Kapitel 6.3.1 beschriebenen Bauteilnetzungen und Roboterbahnen zum Einsatz. Als Footprint diente das aus verschiedenen Footprintversuchen gemittelte Profil $\bar{\mathcal{F}}_{110}$ mit einem Spritzabstand von 110 mm bei einer Spritzdauer von 2,5 s, welches in Abbildung 3.47a dargestellt ist. Zur Berechnung des Haftwirkungsgrads wurde der mittels der in Kapitel 4.4.3 beschriebenen Kalibrierung optimierte Verlauf $\xi_{\text{ang}}(\theta)$ eingesetzt, welcher der durchgehenden orangefarbenen Kurve in Abbildung 4.11 entspricht.

Schichtdickenverteilung

Die Abbildungen 6.14a–6.14d zeigen die gemessenen und simulierten Schichtdickenverteilungen für beide Bauteile. Wie zu erwarten, ist hier zunächst erkennbar, dass die gemessenen Schichtdickenverteilungen verrauscht und die simulierten Verteilungen sehr glatt sind. Die Tatsache, dass bei den simulierten Schichtdickenverteilungen in Bereichen mit weitgehend homogenen geometrischen Eingriffsbedingungen, wie beispielsweise auf der Oberseite der MAT2A-Bauteile, überhaupt sichtbare Schichtdickenschwankungen auftreten, ist darauf zurückzuführen, dass die Verfahrgeschwindigkeit der Spritzpistole Schwankungen unterworfen ist. Diese Schwankungen sind auf die Vorhersage der Roboterdynamik durch Robotstudio zurückzuführen und bedingen stellenweise einen Rückgang der Relativgeschwindigkeit zum Werkstück auf 25 % der Sollgeschwindigkeit für den Bruchteil einer Sekunde. Ob diese einzeln auftretenden Schwankungen eine begründete Vorhersage des Roboterhaltens darstellen, ist zweifelhaft. Wahrscheinlicher erscheinen an dieser Stelle geringfügige numerische Probleme.

Eine weitere Auffälligkeit in der Schichtdickenverteilung von MAT3D in Abbildung 6.14c stellt die berechnete Überhöhung der Schichtdicke in einem näherungsweise kreisförmigen Bereich des Innenradius dar. Diese Überhöhung konnte in der Tat auch in den gemessenen Schichtdickenverteilungen festgestellt werden und ist auf eine komplizierte mehrachsige Umorientierung des Roboters und das damit einhergehende Absinken der Relativgeschwindigkeit des TCP zurückführbar. In der Darstellung der brennerseitigen Roboterbahn auf Seite 243 ist der ursächliche Überschwinger in der Bahn als kleine Schlaufe neben der Spitze des eingezeichneten

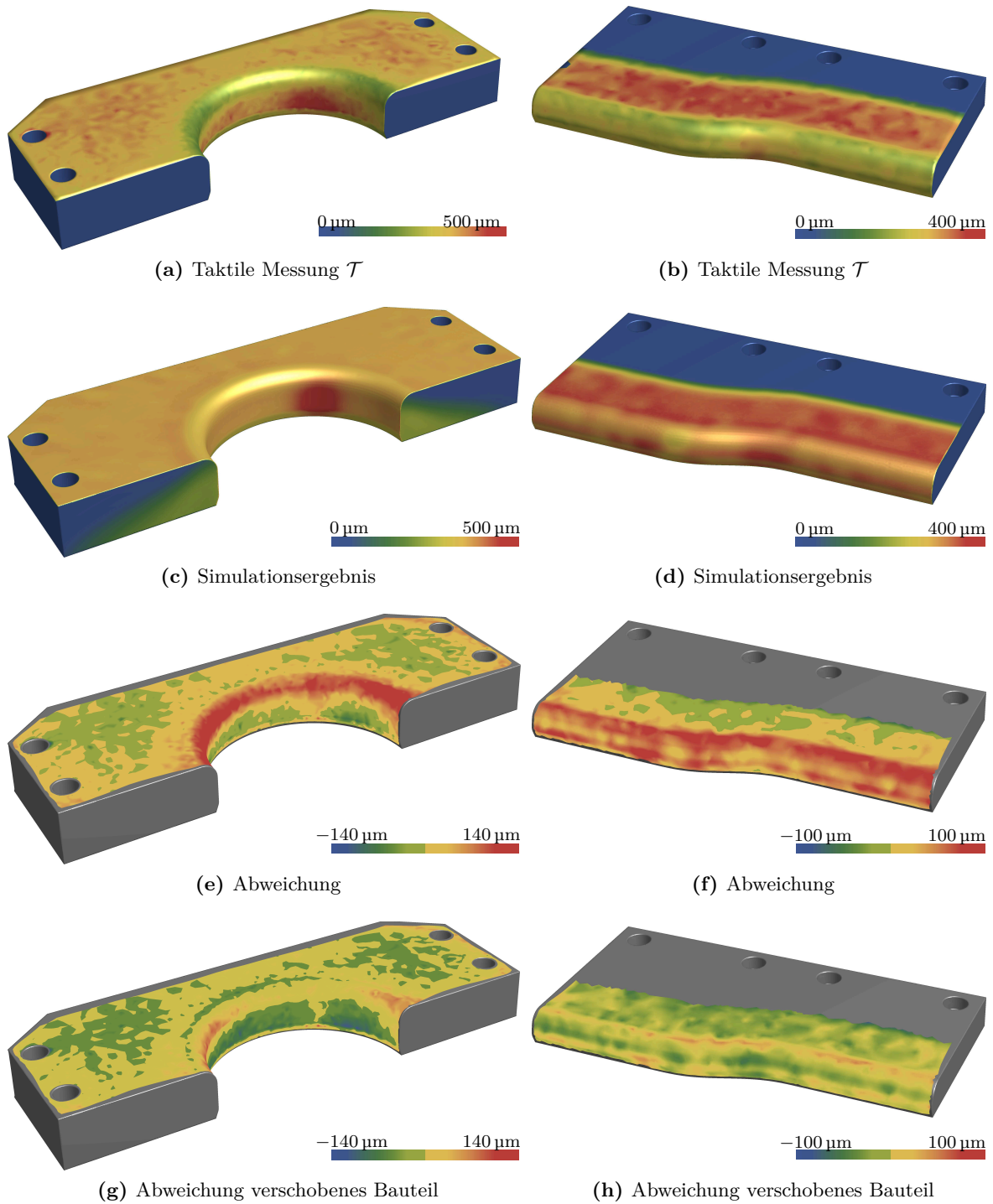


Abbildung 6.14: Vergleich der gemessenen und simulierten Schichtdicken auf den Bauteilen MAT2A und MAT3D. (a) und (b) zeigen die taktile gemessenen Schichtdicken $\mathcal{T}_{2|3}$, (c) und (d) die simulierten Schichtdickenverteilungen mit identischem Maßstab sowie (e) und (f) die vorzeichenbehafteten Abweichungen $S_{2|3}(i) - \mathcal{T}_{2|3}(i)$. (g) und (h) zeigen auf gleiche Weise berechnete Abweichungen, falls eine Verschiebung der Bauteile nach unten (entlang der Normalen der Unterseite) um 2,8mm angenommen wird. Letztere lassen erkennen, dass bereits eine kleine Verschiebung des Bauteils um zirka einen halben Mäanderbahnabstand deutliche Änderungen der Schichtdickenverteilung hervorrufen kann.

orangefarbenen Spritzkegels erkennbar. In der Vorbereitung eines Fertigungsprozesses hätte diese Fehlbeschichtung im Vorhinein simulativ erkannt und kompensiert werden können.

Für die Oberseiten der Bauteile sowie auf dem Innenradiusbereich \mathcal{P}_F von MAT3D stimmen simulierte und gemessene Schichtdicken visuell gut überein, auf den Ziehradien \mathcal{P}_R der Bauteile sind die gemessenen Schichtdicken hingegen geringer als von der Simulation vorhergesagt. Deutlich wird dies insbesondere in den Darstellungen der mittleren Abweichungen $\mathcal{S}_{2|3}(i) - \mathcal{T}_{2|3}(i)$ der simulierten Schichtdickenverteilungen \mathcal{S} von den Messungen \mathcal{T} in den Abbildungen 6.14e und 6.14f.

Für die quantitative Beurteilung der simulierten Schichtdickenverteilung mit Bezug auf die drei in Kapitel 6.3.2 definierten Messergebnisse \mathcal{T} , \mathcal{Q} und \mathcal{C} beinhaltet Tabelle 6.4 die prozentualen und absoluten mittleren Abweichungen der Schichtdicken in den unterschiedlichen Bauteilbereichen. Die Bereiche \mathcal{P}_{Ges} umfassen dabei alle Messpunkte des jeweiligen Bauteils, auch diejenigen in den Übergangsbereichen zwischen bzw. außerhalb der übrigen Segmente⁵. Da die taktilen Scandaten aufgrund der Oberflächenrauheit stark verrauscht und die Simulationsdaten innerhalb eines Bereichs \mathcal{P}_* glatt sind, werden hier die einfachen, vorzeichenbehafteten Fehler betrachtet, welche in dem idealen Fall, dass die Simulationsdaten in einem Segment dem Mittel der gemessenen Schichtdicken entsprechen, Null ergeben. Die Berechnung absoluter oder quadratischer Fehlermaße ist in diesem Fall nicht zweckmäßig, da diese im Wesentlichen die Streuung der verrauschten Messdaten wiedergeben würden. Auch eine Glättung der Messdaten ist an dieser Stelle weniger sinnvoll, da eine ausreichende Glättung auf Basis der bereichsweise relativ weit voneinander entfernten Messpunkte einen zu großflächigen Filterkern erfordern würde. Die im oberen Teil der Tabelle 6.4 angegebenen mittleren Abweichungen berechnen sich mittels

$$E_m(\mathcal{T}, \mathcal{P}_*) = \frac{1}{|\mathcal{P}_*|} \sum_{i \in \mathcal{P}_*} (\mathcal{S}(i) - \mathcal{T}(i))$$

und

$$E_{m\%}(\mathcal{T}, \mathcal{P}_*) = 100 \cdot E_m(\mathcal{T}, \mathcal{P}_*) / \bar{\mathcal{T}}(\mathcal{P}_*),$$

wobei $|\mathcal{P}_*|$ die Anzahl der Messpunkte im Bereich \mathcal{P}_* sowie $\bar{\mathcal{T}}(\mathcal{P}_*)$ den Mittelwert der Messung \mathcal{T} im Bereich \mathcal{P}_* darstellen. Durch Ersetzen von $\mathcal{T}(i)$ durch $\mathcal{C}(i)$ beziehungsweise $\bar{\mathcal{Q}}(\mathcal{P}_*)$, der mittleren Schichtdicke aller Querschliffmesspunkte im Bereich \mathcal{P}_* , ergeben sich analog die Fehlermaße in Bezug auf die korrigierten Schichtdicken und die Querschliffmessungen.

Da die Messpunktdichten in den unterschiedlichen Segmenten der Bauteile nicht konstant sind und sich in den unterschiedlichen Segmenten zudem unterscheidbare Fehlercharakteristiken zeigen, sind im unteren Teil der Tabelle 6.4 flächengewichtete Fehler für das gesamte Bauteil bzw. den gesamten Messbereich angegeben. Es bezeichne dazu $A(i)$ die von einem Vertex i repräsentierte Oberfläche, welche als ein Drittel der Fläche der inzidenten Dreiecke berechnet wird. Für MAT2A beträgt $A(i)$ durchschnittlich $4,5 \text{ mm}^2$ und für MAT3D $7,1 \text{ mm}^2$.

⁵Es ist also für MAT2A $\{\mathcal{P}_O, \mathcal{P}_R\} \subsetneq \mathcal{P}_{Ges}$ und für MAT3D $\{\mathcal{P}_O, \mathcal{P}_R, \mathcal{P}_F, \mathcal{P}_A\} \subsetneq \mathcal{P}_{Ges}$.

Bauteil	Bereich	Messpunkte	Flächenanteil	Mittlere Abweichung der simulierten Schichtdicke					
				\mathcal{T}		\mathcal{Q}		\mathcal{C}	
				$E_m(\mathcal{T}, \mathcal{P}_*)$	$E_{m\%}(\mathcal{T}, \mathcal{P}_*)$	$E_m(\mathcal{Q}, \mathcal{P}_*)$	$E_{m\%}(\mathcal{Q}, \mathcal{P}_*)$	$E_m(\mathcal{C}, \mathcal{P}_*)$	$E_{m\%}(\mathcal{C}, \mathcal{P}_*)$
\mathcal{P}_*	$ \mathcal{P}_* $								
			%	μm	%	μm	%	μm	%
MAT2A	\mathcal{P}_O	1.107	55	3,4	1,0	1,1	0,3	-5,0	-1,4
MAT2A	\mathcal{P}_R	2.215	45	50,7	18,4	41,9	14,7	43,9	15,5
MAT2A	\mathcal{P}_{Ges}	3.322	100	35,0	— ^{a)}	— ^{b)}	— ^{a)}	27,6	— ^{a)}
MAT3D	\mathcal{P}_O	1.699	70.3	10,7	3,0	16,7	4,8	13,2	3,7
MAT3D	\mathcal{P}_R	1.074	13.3	80,1	31,4	58,4	20,9	57,4	20,4
MAT3D	\mathcal{P}_F	1.217	12.1	3,6	1,0	-40,6 ^{c)}	-9,9 ^{c)}	-41,4	-10,7
MAT3D	\mathcal{P}_A	356	4.3	0,9	0,3	-4,7	-1,5	-5,7	-1,8
MAT3D	\mathcal{P}_{Ges}	4.571	100	25,5	— ^{a)}	— ^{b)}	— ^{a)}	8,4	— ^{a)}

Bauteil	Flächen	Flächengewichtete, mittlere Abweichung der simulierten Schichtdicke					
		$\hat{E}_m(\mathcal{T}, \hat{\mathcal{P}})$	$\hat{E}_{m\%}(\mathcal{T}, \hat{\mathcal{P}})$	$\hat{E}_m(\mathcal{Q}, \hat{\mathcal{P}})$	$\hat{E}_{m\%}(\mathcal{Q}, \hat{\mathcal{P}})$	$\hat{E}_m(\mathcal{C}, \hat{\mathcal{P}})$	$\hat{E}_{m\%}(\mathcal{C}, \hat{\mathcal{P}})$
	$\hat{\mathcal{P}}$	μm	%	μm	%	μm	%
MAT2A	$\mathcal{P}_O \cup \mathcal{P}_R$	24,4	8,8	19,5	6,8	16,5	7,7
MAT3D	$\mathcal{P}_O \cup \mathcal{P}_R \cup \mathcal{P}_F \cup \mathcal{P}_A$	17,2	6,4	22,4 ^{d)}	7,1 ^{d)}	10,5	6,7

^{a)} Bezugsschichtdicke nur innerhalb eines Segments definiert. ^{b)} \mathcal{Q} nur innerhalb eines Segments definiert.

^{c)} Werte aufgrund der punktuellen Überbeschichtung auf der Front ungenau. ^{d)} Ohne Frontsegment.

Tabelle 6.4: Mittlere und flächengewichtete mittlere Abweichungen der simulierten Schichtdicke von den Messergebnissen für die verschiedenen Bauteilbereiche. Die Abweichungen sind jeweils in Mikrometern und in Prozent in Bezug auf die taktilen Messungen \mathcal{T} , die durchschnittlichen Querschlifmesswerte \mathcal{Q} und die korrigierten Schichtdickenwerte \mathcal{C} angegeben. Prozentuale Angaben beziehen sich auf die durchschnittliche gemessene Schichtdicke in dem jeweiligen Bereich. Positive Werte bedeuten eine Überschätzung der Schichtdicke durch die Simulation.

Ferner sei

$$A(\mathcal{P}_*) = \sum_{i \in \mathcal{P}_*} A(i)$$

die Fläche eines Segments. Die in Tabelle 6.4 aufgeführte flächengewichtete mittlere Abweichung eines Bauteils für die Menge $\hat{\mathcal{P}}$ der Messpunkte aller Segmente ist

$$E_{mA}(\mathcal{T}, \hat{\mathcal{P}}) = \frac{1}{A(\hat{\mathcal{P}})} \sum_{i \in \hat{\mathcal{P}}} A(i) (\mathcal{S}(i) - \mathcal{T}(i)).$$

Analog ist auch $E_{mA}(\mathcal{C}, \hat{\mathcal{P}})$ definiert. Für die prozentualen Fehler $\hat{E}_{m\%}$ sowie für $\hat{E}_m(\mathcal{Q})$ erfolgt die Flächengewichtung durch einfache Gewichtung der Beträge der im oberen Teil der Tabelle angegebenen Fehlerwerte mit den ebenfalls angegebenen Flächenanteilen der jeweiligen Segmente, da in den Übergangs- und Randbereichen des Bauteils keine einheitliche Bezugsschichtdicke definiert ist. Für den flächengewichteten prozentualen Fehler des Bauteils MAT3D ergibt sich beispielsweise

$$\begin{aligned} \hat{E}_{m\%}(\mathcal{T}) = & 0,703 \cdot |E_{m\%}(\mathcal{T}, \mathcal{P}_O)| + 0,133 \cdot |E_{m\%}(\mathcal{T}, \mathcal{P}_R)| \\ & + 0,121 \cdot |E_{m\%}(\mathcal{T}, \mathcal{P}_F)| + 0,043 \cdot |E_{m\%}(\mathcal{T}, \mathcal{P}_A)|. \end{aligned}$$

Anhand der tabellierten Werte bestätigt sich im Vergleich der Segmente, dass die Schichtdicke insbesondere am Radius auf beiden Bauteilen überschätzt wird. Die Abweichungen reichen dabei klar über das Rauschen der Schichtdickenmessungen hinaus und liegen für MAT2A im Bereich von 14,7–18,4% und für MAT3D im Bereich von 20,4–31,4%. Der Vergleich der präzisen Querschliffmessung mit der simulierten Schichtdicke entlang dieses Querschliffs in Abbildung 6.13 bestätigt dies, zeigt aber auch, dass der sehr hohe Fehler $E_m(\mathcal{T}_3, \mathcal{P}_R)$ von 31,4% unter anderem durch die Unterschätzung der Schichtdicke durch den taktilen Scan zustande kommt. Die Werte $E_m(\mathcal{Q}_3, \mathcal{P}_R)$ sowie $E_m(\mathcal{C}_3, \mathcal{P}_R)$ auf Basis der Querschliffe bzw. der korrigierten taktilen Daten entsprechen daher hier der tatsächlichen Abweichung von ca. 20–21%. In den Fehlerdarstellungen in Abbildung 6.14 ist darüber hinaus ersichtlich, dass die Abweichungen auf dem Ziehradius für MAT2A im Wesentlichen am oberen und unteren Ende des Radius auftreten. Für MAT3D sind die Abweichungen an den Enden des Ziehradius etwas größer als im mittleren Bereich.

Als mögliche technische Ursachen für die erhöhten Abweichungen auf dem Radius kommen Fehlplatzierungen des Bauteils oder Fehleinmessung des Roboters, Abweichungen des Roboters vom programmierten Bahnverlauf oder Prozessfluktuationen in Frage. Insbesondere die Beschichtung des Radiusbereichs des Bauteils MAT3D bringt dabei den eingesetzten Roboter und den montierten Spritzbrenner an seine dynamischen Grenzen, da hier Umorientierungsgeschwindigkeiten des Brenners von mehr als 230°/s erreicht werden müssen. Die dadurch wirkenden extremen Beschleunigungen führen zu ausgeprägten und raumgreifenden Schwingungen des 50 kg schweren Schlauchpaketes, welche mit hoher Wahrscheinlichkeit auch erhöhte Abweichungen im Bahnverlauf hervorrufen. Des Weiteren ist nicht auszuschließen, dass in diesem Fertigungsprozess bereits die statischen durch das Gewicht der Schlauchpakete verursachten Einflüsse zu einer Abnahme der Radiusbeschichtung geführt haben. Dies ist im

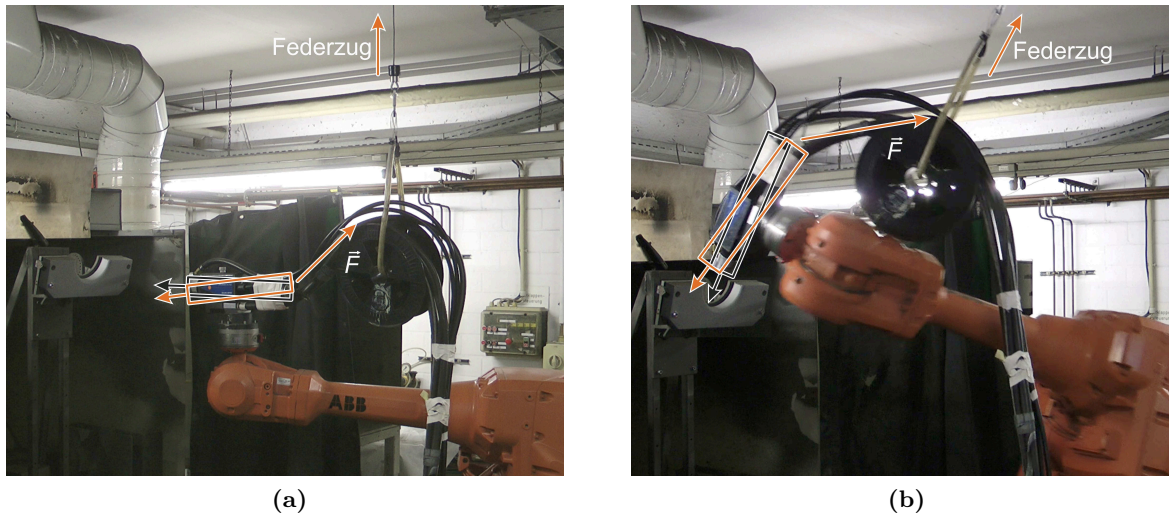


Abbildung 6.15: Krafteinwirkung des Schlauchpakets auf den Brenner bei der Beschichtung des Bauteils MAT3D. (a) Roboterposition zu Beschichtung der Oberseite des mit nach oben offenen Radius eingespannten Bauteils. In diesem Fall wirkt durch das Schlauchpaket, welches über eine aufgehängte Umlenkrolle mit einem starken Federzug geführt wird, eine Kraft \vec{F} , die den Brenner am hinten liegenden Angriffspunkt des Schlauchpaketes nach oben zieht. (b) In der oberen Brennerposition bei der Beschichtung des Frontalbereichs \mathcal{P}_F wirkt die durch das Schlauchpaket ausgeübte Kraft \vec{F} , aufgrund der geänderten Roboterstellung, in entgegengesetzter Richtung. In beiden Fällen resultiert potentiell eine Verkippung des Brenners, welcher den Spritzstrahl vom Ziehradius entfernt.

Versuchsaufbau, welcher in Abbildung 6.15 dargestellt ist, begründet. Da das Bauteil hier so platziert ist, dass der U-förmige Innenradius nach oben zeigt, führen die statischen am Brenner angreifenden Gewichtskräfte des Schlauchpaketes dazu, dass der Spritzstrahl durch Toleranzen und Unsteifigkeiten in der Brennerhalterung, welche bereits in Kapitel 3.4.1 erläutert wurden, vom Radius weggedreht wird. Dies gilt sowohl für die in Abbildung 6.15a gezeigte waagerechte Pose zur Beschichtung der Oberseite \mathcal{P}_O des Bauteils, als auch für die senkrechte „Über-Kopf“-Pose zur Beschichtung des Frontbereichs \mathcal{P}_F (Abbildung 6.15b), welche beide unmittelbar an den Radius angrenzen. Im Vergleich zu der in Kapitel 3.4.1 beschriebenen Halterung des Brenners wurde hier zwar eine verbesserte Version eingesetzt, so dass zumindest für das aus Sicht der Roboterdynamik weniger problematische Bauteil MAT2A die gemessene Abweichung der Brennerausrichtung von bis zu $4,25^\circ$ vermutlich nicht erreicht wird, allerdings hat hier bereits eine minimale beidseitige Abweichung im Bereich von $\pm 0,25^\circ$ eine Verschiebung des TCP um etwa 2,8 mm auf der Oberfläche zur Folge.

Dass bereits Abweichungen von wenigen Millimetern starke Auswirkungen auf die Schichtdickenverteilung haben können, zeigt auch die Durchführung einer automatischen Parameterkalibrierung der Beschichtung für das MAT3D-Bauteil. Diese wurde hier genutzt, um potentielle Fehlplatzierungen oder Fehleinmessungen des Werkstücks im Beschichtungsprozess zu identifizieren, welche die beobachtete Abweichung der Schichtdicke auf dem Ziehradius erklären könnten. In Betracht gezogen wurden dabei Starrkörperverschiebungen und -rotationen ent-

lang der drei Koordinatenachsen bzw. um die drei Koordinatenachsen. Im Rahmen der Optimierung dieser sechs Parameter zur Minimierung der Abweichungen zwischen Simulations- und Messdaten stellte sich heraus, dass zwar von vielen Startpunkten in der Nähe des Nullpunktes (keine Verschiebungen oder Rotationen) eine Verminderung der Abweichungen erzielt werden konnte, die Lösungen jedoch verschiedene lokale Optima darstellen, welche keine eindeutige Aussage über eine optimale Richtung beziehungsweise einen einzelnen signifikanten Parameter zuließen. Daher wurde der Parameterraum für Rotationen um jede Achse im Bereich von -5° bis 5° und für Werkstückverschiebungen im Bereich -5 mm bis 5 mm simulativ abgetastet. Zum Einsatz kamen dabei ein vollfaktorielles 7^6 -Schema sowie 15 Schemata der Größe 101^2 für alle Kombinationen aus je zwei Parametern, so dass insgesamt 270.664 Simulationen durchgeführt wurden. Eine Auswertung der berechneten Daten hat ergeben, dass eine Verschiebung des Bauteils entlang der z-Achse, welche der Normalenrichtung der Oberseite entspricht, den mit Abstand deutlichsten Einfluss auf eine Verminderung der Abweichungen der Simulations- von den Messdaten hat. Eine Verschiebung des Bauteils in andere Richtungen oder Rotationen bewirken hingegen innerhalb des betrachteten Intervalls nahezu keine Veränderung des Fehlers.

Eine exemplarische Verschiebung des Bauteils um $2,8$ mm in Richtung der Bauteiloberseite führt hier zu einer Erhöhung der simulativ abgelagerten Schichtdicke auf dem Radius und dem im Vergleich mit Abbildung 6.14e deutlich verbesserten Fehlerbild in Abbildung 6.14g. Der relative Fehler $E_{m\%}(\mathcal{C}, \mathcal{P}_R)$ sinkt hier von absoluten $20,4\%$ auf $12,3\%$ ab. Der Fehler auf dem Frontsegment \mathcal{P}_F sinkt ebenfalls, von $10,7\%$ auf $6,7\%$, während im Auslaufbereich ein leichter Anstieg auf $6,6\%$ zu verzeichnen ist und die Schichtdicke auf der Oberseite nahezu unverändert bleibt. Der flächengewichtete Gesamtfehler $\hat{E}_{m\%}$ sinkt auf $5,5\%$. Die Abweichungen auf dem Radius sind daher auch durch eine geringfügige Fehlplatzierung des Bauteils oder eine Fehleinmessung des Roboters im Fertigungsprozess erklärbar. Auch für das Bauteil MAT2A verbessert sich durch eine identische Verschiebung der Fehler auf dem Ziehradius, wie Abbildung 6.14h zeigt.

Für die übrigen Bauteilbereiche konnten gemäß Tabelle 6.4 deutlich genauere Vorhersagen der Schichtdicke erreicht werden. Die prozentualen Fehler liegen dabei für die unterschiedlichen Messdatenannahmen \mathcal{T} , \mathcal{Q} und \mathcal{C} im Frontbereich zwischen $1,0\%$ und $10,7\%$ und im Auslaufbereich zwischen $0,9\%$ und $5,7\%$ absolut. Im Hinblick auf die Prozess- und Messunsicherheiten (vergleiche z.B. Abbildungen 3.27 und 3.47b) sind die erreichten Vorhersagegenauigkeiten auf den Oberseiten beider Bauteile, welche prozentuale Abweichungen zwischen $0,3\%$ und $4,8\%$ mit einem Mittel von $2,4\%$ aufweisen, herausragend. Wie in den Darstellungen des Fehlers in den Abbildungen 6.14e und 6.14f erkennbar ist, sind keine systematischen Abweichungen vorhanden und die durch die gewählte Farbskala deutlich sichtbaren Nulldurchgänge des Fehlers verteilen sich entsprechend der in Abbildungen 6.14a und 6.14b dargestellten Messdaten gleichmäßig über die gesamte Oberfläche.

Die im unteren Teil der Tabelle 6.4 dargestellten flächengewichteten Abweichungen für die gesamten Bauteile liegen ebenfalls bei sehr guten $6,8\%$ bis $8,8\%$ für MAT2A und $6,4\%$ bis $7,1\%$ für MAT3D. Die absoluten, in μm angegebenen Fehler liegen im flächengewichteten Durch-

schnitt im Bereich der in Tabelle 6.3 angegebenen Oberflächenrauheiten und Intervallbreiten der Messungen. Mit Ausnahme der Ziehradienbereiche gilt dies auch für die überwiegende Mehrheit der Einzelsegmentfehler.

Sekundäre Zielgrößen

Abgesehen von der Schichtdicke werden von der Beschichtungssimulation weitere, sekundäre Zielgrößen auf der Bauteiloberfläche und für die Positionen der Roboterbahn berechnet, welche auf den gemeinsamen Grundgrößen Spritzabstand und Spritzwinkel beruhen. Da es sich hierbei um geometrische Größen handelt, deren Berechnung im Gegensatz zur Schichtdickenberechnung nicht auf einem Prozessmodell beruht, ist eine Validierung dieser Größen anhand des realen Prozesses nicht notwendig. Die Verifikation der Berechnungen, also die Prüfung der Korrektheit der Implementierung, erfolgte zum Einen anhand einfacher Simulationsbeispiele wie in Kapitel 6.1 und zum Anderen durch die kontinuierliche Überprüfung der Plausibilität berechneter Werte für eine Vielzahl durchgeführter Simulationen. Im Folgenden werden einige exemplarische Ergebnisse der Berechnung sekundärer Zielgrößen für das Bauteil MAT3D vorgestellt.

Abbildung 6.16a zeigt zunächst den an jedem Punkt auf der Bauteiloberfläche auftretenden maximalen Spritzwinkel im Bereich 0–90°. Das Minimum des maximalen Spritzwinkels beträgt ca. 8° und tritt auf einem großen Teil der Oberseite (blauer Bereich) auf. Da hier die Anstellung des Brenners in der Regel senkrecht ist, entspricht hier der maximale Spritzwinkel dem maximalen Öffnungswinkel des Kernstrahlbereichs (vergleiche Kapitel 5.5.1) des Spritzstrahls. Auf einem Teil der Oberfläche beträgt der maximale Spritzwinkel jedoch fast 90°. Dies ist darin begründet, dass einer der Bahnabschnitte, welcher den Radius mit geneigtem Brenner beschichtet, auch die Oberfläche überstreift. Abbildung 6.16b zeigt den virtuellen Spritzkegel auf dem verursachenden Bahnabschnitt sowie die zu diesem Simulationszeitpunkt vorherrschenden maximalen Spritzwinkel auf der Bauteiloberfläche. Die flache Beschichtung der Oberseite kann, insbesondere bei mehreren Beschichtungsüberläufen, zu einer rauen Oberfläche und einer insgesamt schlechten Schichtqualität führen. Das Erkennen dieser Situation in der Simulation ermöglicht hier das Umgehen dieser Problematik durch eine geringfügige Anpassung der Brenneranstellung in dem kritischen Bahnabschnitt. Auch der in Abbildung 6.16c dargestellte maximale Spritzabstand zeigt deutlich das flach bestrichene Gebiet auf der Oberseite, welches hier durch überdurchschnittliche Spritzabstände heraussticht. Darüber hinaus zeigt Abbildung 6.16d die bahnbezogenen maximalen Spritzabstände entlang der Roboterbahn, welche hier auf der Bauteiloberfläche dargestellt ist. Darin ist der für das flache Überstreichen der Oberfläche verantwortliche Bahnabschnitt auf dem Radius gut identifizierbar. Die Musterung der auf der Bauteiloberseite dargestellten Bahnen gibt hier die Konturen des Bauteilrandes und der Bohrungen wieder, da an diesen Stellen der Spritzabstand durch die Messpunkte in den Bohrungen und auf den Seitenflächen des Bauteils ansteigt.

Im Gegensatz zu den maximalen Abständen und Winkeln zeigen die berechneten durchschnittlichen Spritzabstände und Winkel in den Abbildungen 6.16e und 6.16f ein deutliche homoge-

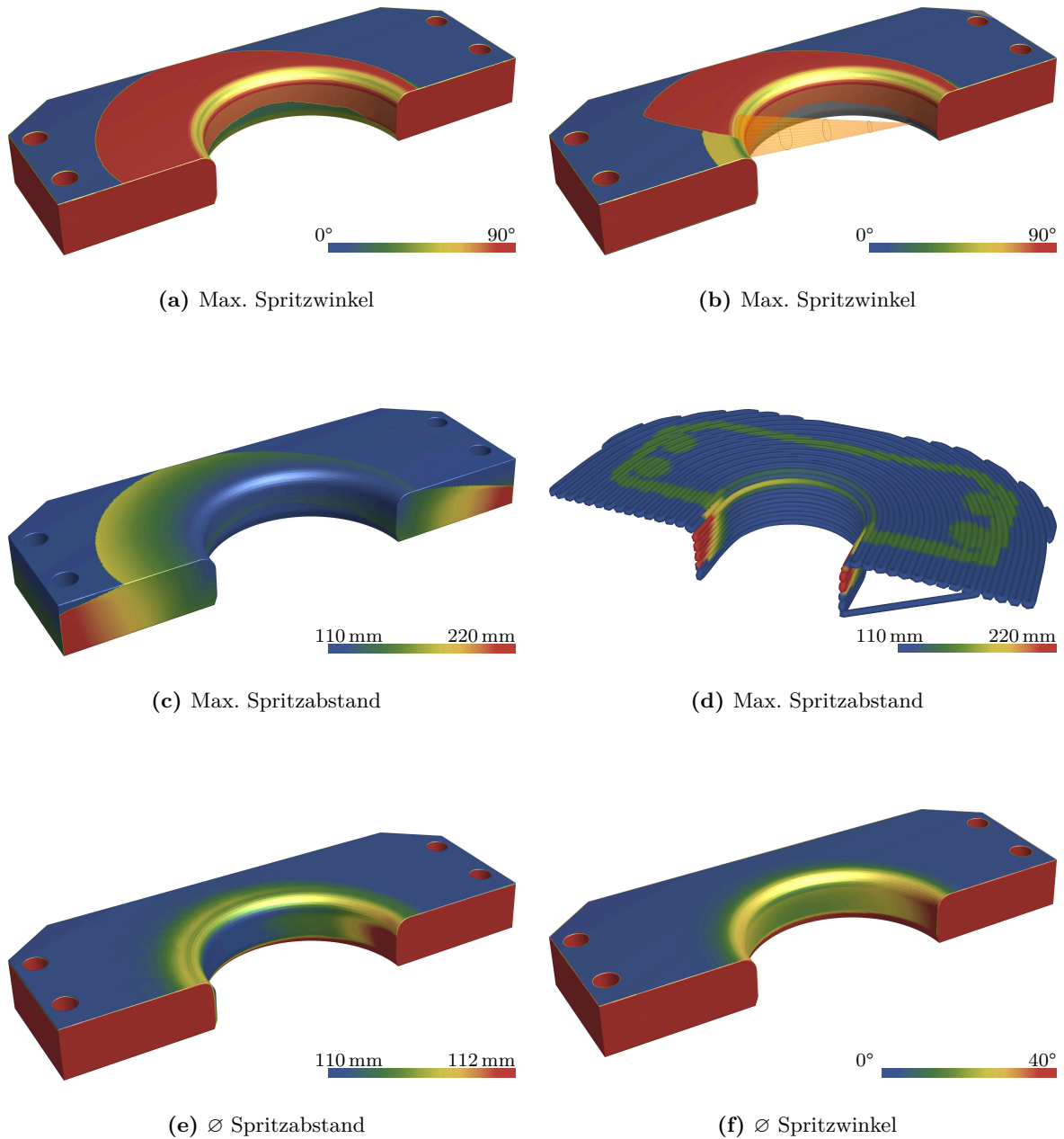


Abbildung 6.16: Simulierte sekundäre Oberflächenzielgrößen für das Bauteil MAT3D. Der maximale Spritzwinkel in (a) und (b) sowie der maximale Spritzabstand (c) lassen ein ungewolltes Beschichten der Oberseite von einem Bahnabschnitt am Ziehradius erkennen. Der bahnbegogene maximale Spritzabstand (d) identifiziert dieses Bahnteilstück. Die durchschnittlichen Abstände (e) und Winkel (f) sind homogener mit leichten bis mittleren Abweichungen von den angestrebten Werten in den Radius- und Frontbereichen \mathcal{P}_R und \mathcal{P}_F .

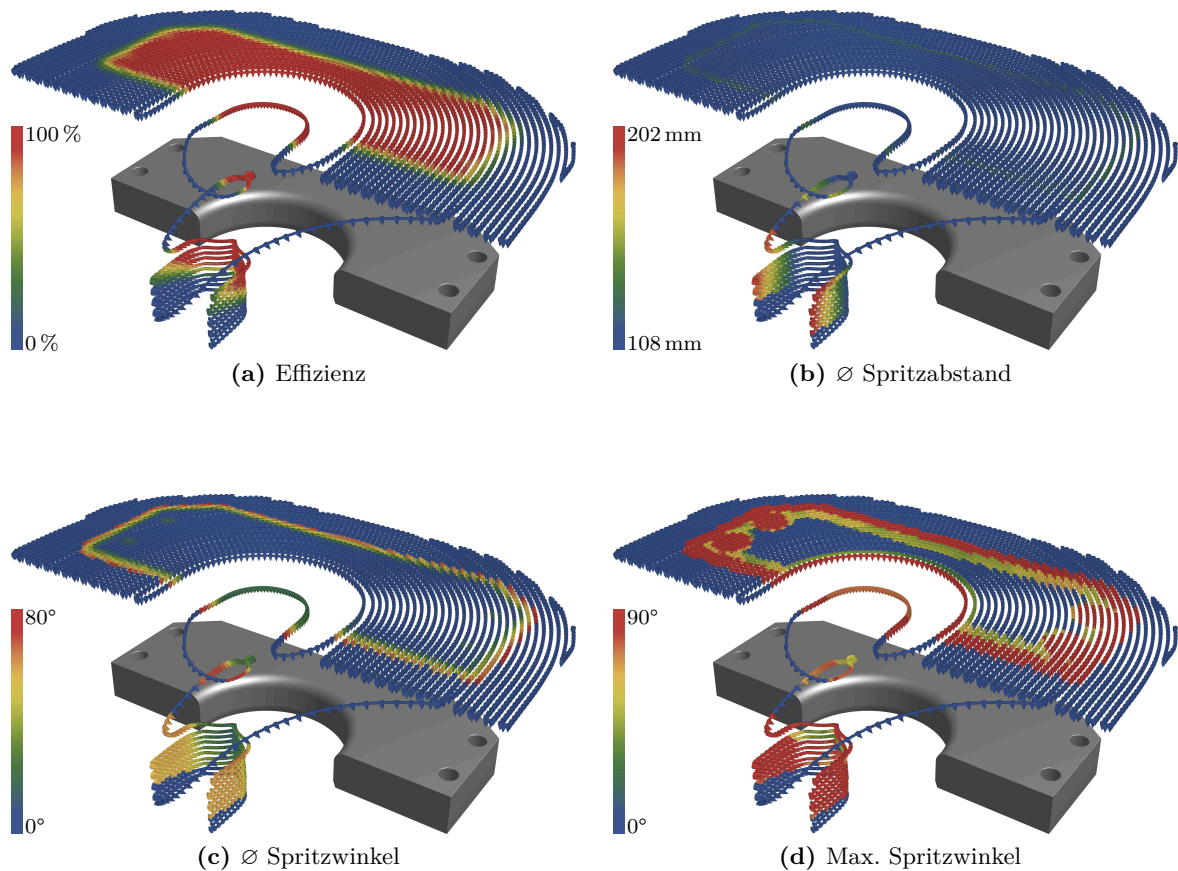


Abbildung 6.17: Ausgewählte simulierte Bahnzielgrößen für das Bauteil MAT3D.

neres Bild. Der Darstellungsbereich der Farbskala für den durchschnittlichen Spritzabstand liegt zwischen 110 mm und 112 mm und zeigt, dass der durchschnittliche, volumengewichtete Abstand auf dem Ziehradius maximal 1 mm und an der rechten Seite des Frontbereichs etwa 2 mm vom angestrebten Standardabstand für den Lichtbogenspritzprozess abweicht. Der durchschnittliche Spritzwinkel beträgt auf dem Ziehradius 20–24° und erreicht seine Maxima innerhalb der Funktionsfläche an den beiden Außenseiten des Frontbereichs \mathcal{P}_F mit etwa 40°. Diese Maxima belegen die Tatsache, dass das Erreichen senkrechter Anstellungen hier nicht möglich ist, da der Brenner sonst auf der jeweils gegenüberliegenden Seite des Radius mit dem Bauteil kollidieren würde.

Die in Abbildung 6.17 dargestellten bahnbezogenen Zielgrößen erleichtern insbesondere im Zusammenspiel mit den oberflächenbezogenen Werten eine einfachere Interpretation des geplanten Beschichtungsprozesses und dienen als Grundlage zur Identifizierung von potentiell problematischen Bahnabschnitten. Die in Abbildung 6.17a dargestellte Prozesseffizienz, also der Anteil des abgelagerten Materials am Gesamtmaterial, zeigt dabei auf einfache Weise an, ob die Roboterbahn an allen Seiten des Bauteilrandes weit genug verlängert wurde, so dass sich der Brenner zum Zeitpunkt des Wendevorgangs nicht mehr im Eingriff befindet.

Umgekehrt ist auch erkennbar, ob am Bauteilrand eventuell einige Mäanderabschnitte der Bahn gekürzt werden können, um Material zu einzusparen und die Prozesskosten zu senken. In Abbildung 6.17a ist ersichtlich, dass hier beispielsweise zwei bis vier Mäanderabschnitte eingespart werden könnten.

Wie bereits bei den oberflächenbezogenen Zielgrößen gezeigt wurde, sind der durchschnittliche Spritzabstand und -winkel sehr homogen und auch die bahnbezogenen Werte zeigen hier keine gravierenden Ausreißer. Zwar zeigt Abbildung 6.17b an den Umkehrpunkten der Bahn über dem Radius des Bauteils erhöhte Spritzabstände, diese sind jedoch auf die Beschichtung der nicht zur Region of Interest gehörenden Stirnflächen zurückzuführen (vergleiche Abbildung 6.16c). Im Vergleich mit der bahnbezogenen Prozesseffizienz ist ferner ersichtlich, dass in den Bahnabschnitten mit hohem Spritzabstand kaum Schichtauftrag stattfindet. Die bahnbezogenen Spritzwinkel zeigen erwartungsgemäß die unvermeidbar erhöhten Werte im Bereich der Ziehradiusbeschichtung (Abbildung 6.17c). Die Darstellung der maximalen Spritzwinkel in Abbildung 6.17d bildet wiederum die Bauteilkonturen und Bohrungen ab und zeigt, dass auch die Frontseite \mathcal{P}_F vom innersten radialen Bahnsegment der Oberseite unter sehr flachen und damit ungünstigen Winkeln beschichtet wird. Wie auch schon beim Überstreichen der Oberseite von einer der Radiusbahnen aus bestünde hier die Möglichkeit, dies durch ein minimales Ankippen der Brennerorientierung vor dem Erreichen des Radius zu unterbinden und somit die resultierende Oberflächenqualität auf dem Frontbereich positiv zu beeinflussen.

6.4 Sensitivitätsanalyse und simulationsgestützte Bahnoptimierung

In den folgenden beiden Abschnitten wird die Nutzung der Simulation zur Sensitivitätsanalyse (vergleiche Kapitel 5.8.2) und zur Bahnoptimierung exemplarisch demonstriert.

6.4.1 Sensitivitätsanalyse für Tiefziehwerkzeuge

Neben der Bahnoptimierung stellt die in Kapitel 5.8.2 vorgestellte Methode zur One-at-Time Sensitivitätsanalyse einen Anwendungsfall dar, bei dem die Beschichtungssimulation zur Einschätzung kritischer und unkritischer Einmessparameter genutzt wird. Für die beiden betrachteten Bauteile MAT2A und MAT3D wurde jeweils eine Sensitivitätsanalyse in Bezug auf Verschiebungen und Rotationen entlang bzw. um die drei x-, y- und z-Achsen der Bauteile durchgeführt, welche in den Darstellungen im Anhang definiert sind (Seiten 242 und 244). Abbildung 6.18 zeigt die resultierenden Verläufe der durchschnittlichen absoluten Abweichung der Beschichtung des rotierten oder verschobenen Bauteils gegenüber der Beschichtung in ursprünglicher Bauteilposition, wofür 100 Simulationen pro Translations- oder Rotationsparameter in 0,1 mm- bzw. 0,1°-Schritten im Bereich ± 5 mm respektive $\pm 5^\circ$ durchgeführt wurden. Auffallend ist, dass die mit zunehmender Verschiebung oder Rotation des Bauteils ansteigenden Abweichungen für das MAT2A-Bauteil größer sind als die für MAT3D. Dies ist darin begründet, dass die Roboterbahn für MAT2A verkürzt wurde und über der Bauteilo-

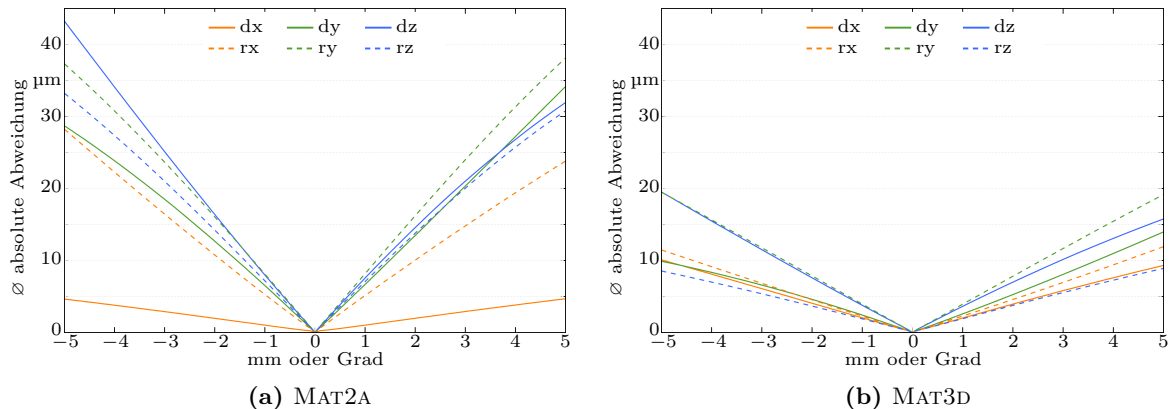


Abbildung 6.18: Untersuchung der Sensitivität der Beschichtung gegenüber Verschiebungen dx , dy und dz sowie Rotationen rx , ry und rz um die durch den Mittelpunkt der Bauteile verlaufenden Bauteilachsen. Als Maß dient hier die durchschnittliche absolute Abweichung der Schichtdickenverteilungen von der der Ausgangskonfiguration mit $dx = dy = dz = 0$ und $rx = ry = rz = 0$. (a) Sensitivitätswerte für MAT2A. (b) Sensitivitätswerte für MAT3D.

berfläche ausläuft, so dass in diesem Übergangsbereich schon bei geringen Translationen oder Rotationen deutliche Schichtdickenänderungen auftreten, wie in Abbildung 6.19a anhand einer Rotation um 5° um die z-Achse dargestellt ist. Eine dementsprechend geringe Sensitivität ergibt sich für Verschiebungen entlang der x-Achse, da hierbei der Auslaufbereich kaum verändert wird. Für das Bauteil MAT3D treten vergleichbare Schichtdickenschwankungen nicht auf, da hier die Bahnen über den Bauteilrand hinausragen.

Da die Verläufe der Sensitivitätskurven in der Nähe der Ausgangskonfiguration näherungsweise linear sind, können für diesen Bereich Sensitivitätswerte in Mikrometern durchschnittlicher Schichtdickenänderung pro mm Bauteilverschiebung beziehungsweise pro Grad Bauteilrotation angegeben werden. Das Diagramm 6.19b fasst diese für beide Bauteile zusammen. Dabei sind Sensitivitätswerte für negative Verschiebungen und Rotationen nach unten aufgetragen und zusätzlich zu den durchschnittlichen absoluten Fehlern auch die maximalen absoluten Fehler pro Millimeter oder Sekunde als schmalere Balken dargestellt⁶. Auftretende Sensitivitäten liegen für den durchschnittlichen Fehler zwischen $1,8 \mu\text{m}/\text{mm}$ und $8,3 \mu\text{m}/\text{mm}$ beziehungsweise $1,8 \mu\text{m}/^\circ$ und $8,0 \mu\text{m}/^\circ$. Für den maximalen Fehler ergeben sich Werte zwischen $6,5 \mu\text{m}/\text{mm}$ und $24 \mu\text{m}/\text{mm}$ beziehungsweise $13,7 \mu\text{m}/^\circ$ und $41,8 \mu\text{m}/^\circ$. Im Vergleich mit der zwischen $300 \mu\text{m}$ und $400 \mu\text{m}$ liegenden Schichtdicke sind dies Maximalfehler von über 10% pro Millimeter oder Grad Abweichung. Dies unterstreicht die wesentliche Bedeutung einer exakten Einmessung des Bauteils und einer exakten Roboterführung im Prozess. In Kombination mit den in der Simulationsumgebung anzeigbaren, bauteilüberdeckenden Abweichungen für verschiedene Rotationen oder Verschiebungen wird mittels der Sensitivitätsanalyse eine Einschätzung der besonders betroffenen Bauteilbereiche und damit eine verbesserte Fertigungsplanung ermöglicht.

⁶Auch die Entwicklung der maximalen Fehler ist in diesem Bereich näherungsweise linear.

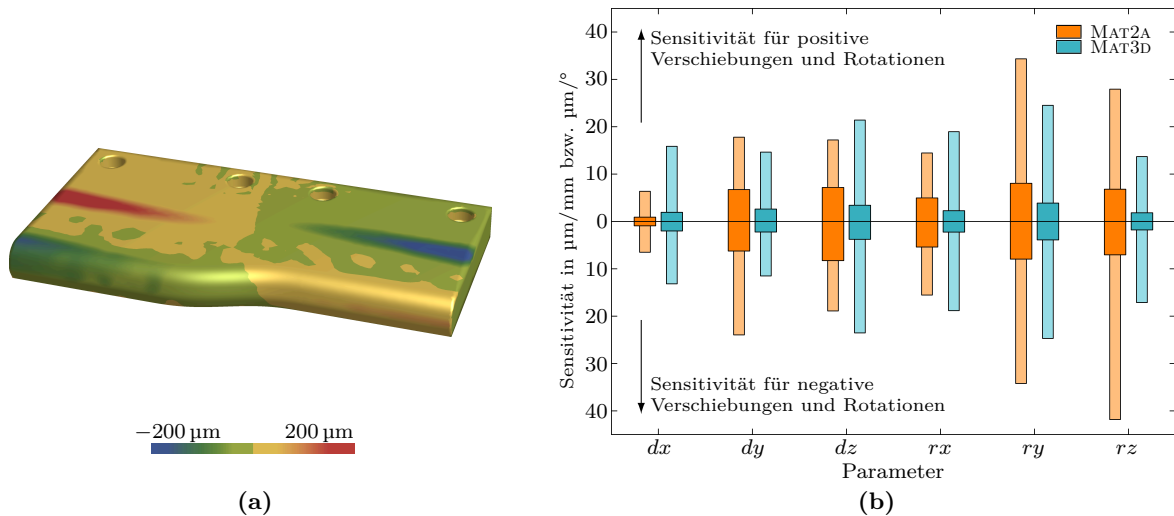


Abbildung 6.19: (a) Abweichungen der Schichtdicke durch eine Rotation von 5° um die senkrecht zur Oberseite stehende z -Achse. Aufgrund der auf der Bauteiloberseite auslaufenden Beschichtung resultieren hier besonders große Schichtdickenabweichungen. (b) Zu erwartende durchschnittliche (breite Balken) und maximale (schmale Balken) Schichtdickenabweichungen in μm pro mm Verschiebung (dx , dy , dz) oder pro Grad Rotation (rx , ry , rz) für die Bauteile MAT2A und MAT3D.

6.4.2 Einsatz der Simulation zur Bahnoptimierung

Um die praktische Nutzbarkeit der entwickelten Beschichtungssimulation für den Einsatz in der iterativen Bahnoptimierung zu demonstrieren, wurden die von Hegels et al. [71] entwickelten Optimierungsalgorithmen an das Netzwerkinterface der Beschichtungssimulation angebunden. Diese Algorithmen gehen von einer bereits existierenden Roboterbahn aus und optimieren diese mit dem Ziel, eine vorgegebene konstante Schichtdicke in der Region of Interest zu erzielen. Der Parameterraum der Optimierung umfasst dabei die Positionen und Ausrichtungen der diskreten Roboterkonfigurationen $\mathbf{p}_i = (\mathbf{x}_i, \mathbf{q}_i, t_i)$ der Bahn sowie die Bahngeschwindigkeit, welche durch die Zeitpunkte t_i kodiert wird (vergleiche Kapitel 5.3.2). Als Optimierungsverfahren kommt dabei ein nichtlineares CG-Verfahren zum Einsatz [134]. Die Berechnung der erforderlichen Gradienten erfolgt darin mittels eines einfachen Ersatzmodells für die Beschichtungsberechnung, welches auf Basis der in dieser Arbeit entwickelten Beschichtungssimulation kalibriert wird, um den im Allgemeinen nicht normalverteilten Footprint zu approximieren. Die Auswertung der Zielfunktion erfolgt in jeder Iteration des Verfahrens durch die exaktere, im Rahmen dieser Arbeit entwickelte Simulation.

Die Anwendungsgebiete für eine Bahnoptimierung, bei der das Vorhandensein einer initialen Bahn vorausgesetzt wird, umfassen die Anpassung eines Prozesses auf eine neue Zielschichtdicke, auf geänderte Spritzverfahren oder -parameter sowie auf geringfügige Änderungen der Bauteilgeometrie. Eine simulationsbasierte Bahnoptimierung erspart in diesen Fällen eine potentiell aufwendige Planung einer gänzlich neuen Bahn. Zur Demonstration dieses simulationsgestützten Optimierungsverfahrens wurde die in dieser Arbeit zur Beschichtung des

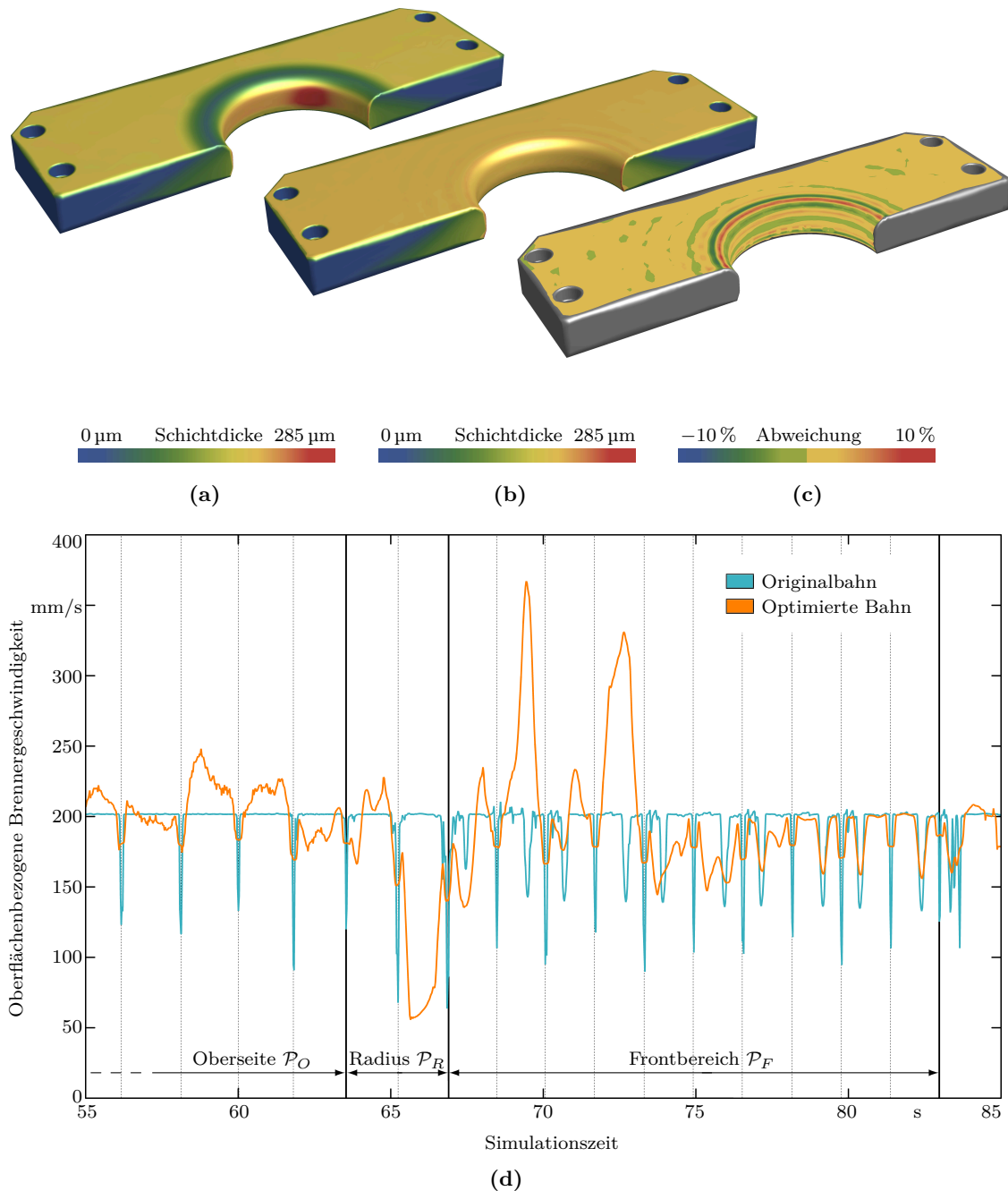


Abbildung 6.20: Anpassung der Roboterbahn für das Bauteil MAT3D auf das Matrizensegment MAT3C durch simulationsbasierte Bahnoptimierung. (a) Beschichtung des Bauteils MAT3C mit der originalen Bahn. (b) Beschichtung mit der optimierten Bahn. (c) Prozentuale Abweichungen der Schichtdicke von der Zielschichtdicke 175 μm für die optimierte Bahn. (d) Verlauf der originalen und der optimierten oberflächenbezogenen Brennergeschwindigkeit. Senkrechte Linien geben die Umkehrpunkte des Brenners außerhalb der Bauteiloberfläche an. Die Anpassung der Geschwindigkeiten auf und in der Nähe des Radius tragen wesentlich zum Erreichen der homogenen Zielschichtdicke bei.

Bauteils MAT3D verwendete und in Anhang D gezeigte Roboterbahn für die Beschichtung des Matrizensegments MAT3C adaptiert, welches dem Segment MAT3D in der Tiefziehmatrize gegenüber liegt (siehe Seite 242 oben rechts). Abgesehen von der kleineren Segmenthöhe besteht ein wesentlicher Unterschied in dem für das Segment MAT3C mit $R = 50\text{ mm}$ im Vergleich zu $R = 65\text{ mm}$ bei MAT3D-Segment deutlich geringeren Radius des Frontausschnitts. Dementsprechend resultiert für das MAT3C Bauteil mit der unveränderten Roboterbahn des MAT3D-Bauteils zunächst die in Abbildung 6.20a dargestellte Fehlbeschichtung. Auf der Oberseite in der Nähe des Ziehradius ist die Beschichtung deutlich zu niedrig, da dieser Teil der Oberfläche bei MAT3D nicht existiert. Im Frontbereich des Werkstücks resultiert eine deutlich zu hohe Schichtdicke aufgrund des in diesem Bereich zu geringen Abstandes des Brenners von der Oberfläche. Nach 50 Iterationen der iterativen Bahnoptimierung mit der Zielvorgabe einer konstanten Schichtdicke von $175\text{ }\mu\text{m}$ resultiert die in Abbildung 6.20a dargestellte Beschichtung, welche die Abweichungen größtenteils behebt. Dies wird durch geringfügige Korrekturen der Brennerausrichtung entlang der Bahn sowie durch die in Abbildung 6.20d dargestellten Veränderungen der Bahngeschwindigkeit erreicht. In dieser Abbildung stellen die senkrechten Orientierungslinien die Umkehrpunkte des Brenners außerhalb des Bauteils dar. Die fettgedruckten Linien entsprechen darin von links nach rechts dem Übergangspunkt zwischen Oberseite und Radius, dem Übergangspunkt zwischen Radius und Front sowie dem Ende der Beschichtung des Frontbereichs \mathcal{P}_F . Es ist zu erkennen, dass insbesondere die Geschwindigkeit der Bahnmäander, welche für das Bauteil MAT3D entlang des Radius verlief und nun die Oberseite nahe des verkleinerten radialen Ausschnitts beschichtet, stark verlangsamt wurde, um die in Abbildung 6.20a fehlende Beschichtung auszugleichen. Des Weiteren sind die Überlaufgeschwindigkeiten auf dem Radius beschleunigt, so dass die Schichtdicke auf dem Radius reduziert wird. Auch die punktuelle Überbeschichtung auf der Bauteilfront wurde so behoben.

Abbildung 6.20a zeigt den prozentualen Schichtfehler gegenüber der Zielschichtdicke. Wie erwartet sind die deutlichsten Restabweichungen am Radius sichtbar, welcher im Vergleich zum Bauteil MAT3D um 15 mm verkleinert wurde. Die maximale Abweichung von der Zielvorgabe beträgt hier bis zu $8,7\%$, im Mittel beträgt der Fehler $1,63\%$. Im Vergleich zur Ausgangssituation mit einem durchschnittlichen Fehler von $18,1\%$ stellt dies einen sehr guten Wert dar.

Zusammenfassung

Im Rahmen dieser Arbeit wurde eine GPU-basierte Beschichtungssimulation für robotergestützte thermische Spritzprozesse konzipiert und realisiert, welche die effiziente Berechnung der Schichtdickenverteilung auf komplexen dreidimensionalen Bauteiloberflächen ermöglicht. Zu diesem Zweck wurde ein Footprint-basiertes Beschichtungsmodell entworfen, welches die Modellierung unterschiedlicher Spritzcharakteristiken und -prozesse auf Basis experimentell ermittelter Schichtprofile erlaubt.

Der erste Teil dieser Arbeit befasste sich daher ausführlich mit der experimentellen Erzeugung, der Digitalisierung und der Nachbearbeitung von Footprintprofilen, deren exakte Erfassung maßgeblich für die Genauigkeit der Beschichtungssimulation ist. Dabei wurde zunächst die grundlegende experimentelle und algorithmische Vorgehensweise dargestellt. Im Folgenden wurden verschiedene potentielle Probleme in der Herstellung und Vermessung der benötigten Schichtprofile identifiziert und technische sowie algorithmische Lösungen vorgestellt. Zur genauen und wiederholbaren Einmessung der Spritzpistole auf das zu beschichtende Bauteil wurde eine unmittelbar auf der Spritzpistole montierbare Lasereinmessvorrichtung konstruiert, welche die Kompensation von translatorischen und rotatorischen Abweichungen ermöglicht und ein Referenzkoordinatensystem für die bisher geometrisch schwer erfassbare Spritzrichtung im Lichtbogenspritzprozess definiert. Zur Digitalisierung der sehr dünnen Schichtprofile wurden sechs taktile und optische 3D-Scansysteme unterschiedlicher Bau- und Funktionsweise untersucht und miteinander sowie mit lichtmikroskopischen Vermessungsmethoden verglichen. Zur digitalen Nachbearbeitung der Scandaten wurden zwei Methoden zur Kompensation thermisch bedingter Verformungen des Substrats vorgestellt und verglichen, durch die eine Verbesserung der Profiligenauigkeit erreicht werden konnte. Ferner wurde der Einfluss des zur Probenpräparation eingesetzten Sandstrahlprozesses auf die Messungen mit taktilen Systemen untersucht und als nicht zu vernachlässigender Faktor für die Messgenauigkeit

identifiziert. Um darüber hinaus auch die vergleichsweise hohe Genauigkeit zweidimensionaler lichtmikroskopischer Messungen in metallographischen Querschliffen für eine Kalibrierung des dreidimensionalen Beschichtungsmodells nutzen zu können, wurde ein modellbasiertes Optimierungsverfahren entwickelt und beschrieben. Dieses Verfahren erlaubt die geometrische Optimierung der dreidimensionalen Footprintform auf Basis mehrerer Querschnitte von Schichtprofilen, die mittels linearer Verfahrensbahnen erzeugt wurden. Abschließend wurden in diesem ersten Teil der Arbeit Methoden zur Darstellung von Footprintprofilen durch die Überlagerung bivariater Gaußfunktionen untersucht. Im Gegensatz zu bisherigen aus der wissenschaftlichen Literatur bekannten Ansätzen wurde dabei die Anzahl der Gaußfunktionen nicht auf die Anzahl der Partikelinjektoren beschränkt. Auf diese Weise konnten die Vorteile funktionaler Repräsentationen, gute Filtereigenschaften und eine kontinuierliche Repräsentation der Massenstromdichte im Randbereich, mit der Fähigkeit einer numerischen Repräsentation, nahezu beliebige Spritzcharakteristiken abbilden zu können, kombiniert werden.

Der zweite Teil dieser Arbeit befasste sich mit dem Entwurf des dreidimensionalen Footprintbasierten Beschichtungsmodells und des GPU-beschleunigten Simulationssystems. Das Beschichtungsmodell wurde in drei logische Teile gegliedert. Der erste Teil beschreibt eine grundlegende, den Prozess definierende Massenstromdichteverteilung auf Basis der ermittelten Footprintprofile. Der zweite Teil des Modells definiert eine geometrische Übertragungsfunktion, welche die Umrechnung zwischen Footprintexperiment und allgemeinen im Rahmen der Simulation auftretenden Eingriffssituationen erlaubt. In diesem Kontext wurden Übertragungsfunktionen für kegelförmige und zylindrische Strahlformen hergeleitet, veranschaulicht und mit einer vereinfachten Formulierung verglichen. Der dritte Teil des Modells bildet den variablen Haftwirkungsgrad ab, zu dessen Bestimmung eine automatisierte, simulationsgestützte Verfahrensweise vorgestellt wurde.

Das Simulationskonzept zeichnet sich insbesondere durch die Abbildung geometrischer Gegebenheiten des Schichtablagerungsprozesses auf den kamerabasierten Bildgenerierungsprozess der OpenGL-Renderingpipeline aus. Dadurch wurde die Ausnutzung der Rechenleistung moderner Grafikkarten zur Schichtberechnung ermöglicht. Eine hohe Simulationsgeschwindigkeit, welche insbesondere für die Verwendung der Beschichtungssimulation in automatisierten Bahnplanungs- und Bahnoptimierungssystemen notwendig ist, konnte durch die Implementierung weiterer Teile der Simulation in Form von GLSL-Shaderprogrammen erzielt werden. Die interaktive Visualisierung der Schichtdickenverteilung und weiterer oberflächen- und bahnbbezogener Zielgrößen sowie die vorgestellten Verfahren zur automatisierten Parameterkalibrierung und zur Sensitivitätsanalyse erlauben eine detaillierte Beurteilung und Planung eines Beschichtungsprozesses. In der ausführlichen Evaluation wurde die Beschichtungssimulation anhand einfacher Experimente verifiziert und anhand der Beschichtung komplexer Tiefziehwerkzeuge validiert. Im Kontext der Beschichtung von Tiefziehwerkzeugen wurden darüber hinaus eine Sensitivitätsanalyse durchgeführt und der Einsatz der Simulation im Rahmen eines automatisierten Bahnoptimierungsverfahrens demonstriert. Ein Vergleich mit einer Implementierung auf Basis der GPU-Bibliothek Optix Prime von Nvidia bestätigte die hervorragenden Laufzeiteigenschaften des in dieser Dissertation konzipierten Simulationssystems.



Flächeninhalt eines elliptischen Kegelschnitts

Gegeben sei zunächst ein einseitiger Kreiskegel K mit Öffnungswinkel 2γ sowie eine Ebene E , welche die Symmetrieachse von K im Abstand d von der Kegelspitze im Punkt D unter dem Winkel θ zur Ebenennormalen schneidet (Abbildung A.1b). Gesucht ist im Folgenden der Flächeninhalt der Schnittellipse. Parametrisierungen, für die die Schnittmenge keine Ellipse darstellt, werden nicht betrachtet.

Da der Flächeninhalt einer Ellipse mittels $A = \pi ab$ aus den Längen a und b der beiden Halbachsen bestimmt werden kann, genügt es zunächst, deren Längen zu bestimmen. Für die Berechnung wird das dreidimensionale Koordinatensystem so gewählt, dass

- der Ursprung in der Kegelspitze liegt,
- die z-Achse der Kegelachse entspricht und
- die Neigung θ von E in der x-z-Ebene gegeben ist; E also aus der Ebene $z = d$ durch Rotation um eine zur y-Achse parallele, durch D verlaufende Rotationsachse hervorgeht.

Die Berechnung der langen Halbachse $a = \frac{1}{2}\overline{S_1S_2}$ kann daher in der x-z-Ebene, welche in Abbildung A.1a dargestellt ist, durchgeführt werden. Mit dem Sinussatz folgt für die zwei Teilstrecken von $\overline{S_1S_2}$

$$\overline{S_1D} = h \cdot \frac{\sin \gamma}{\sin \delta} = d \cdot \frac{\sin \gamma}{\sin(\frac{\pi}{2} - \gamma - \theta)}$$

sowie

$$\overline{DS_2} = h \cdot \frac{\sin \gamma}{\sin \epsilon} = d \cdot \frac{\sin \gamma}{\sin(\frac{\pi}{2} - \gamma + \theta)}$$

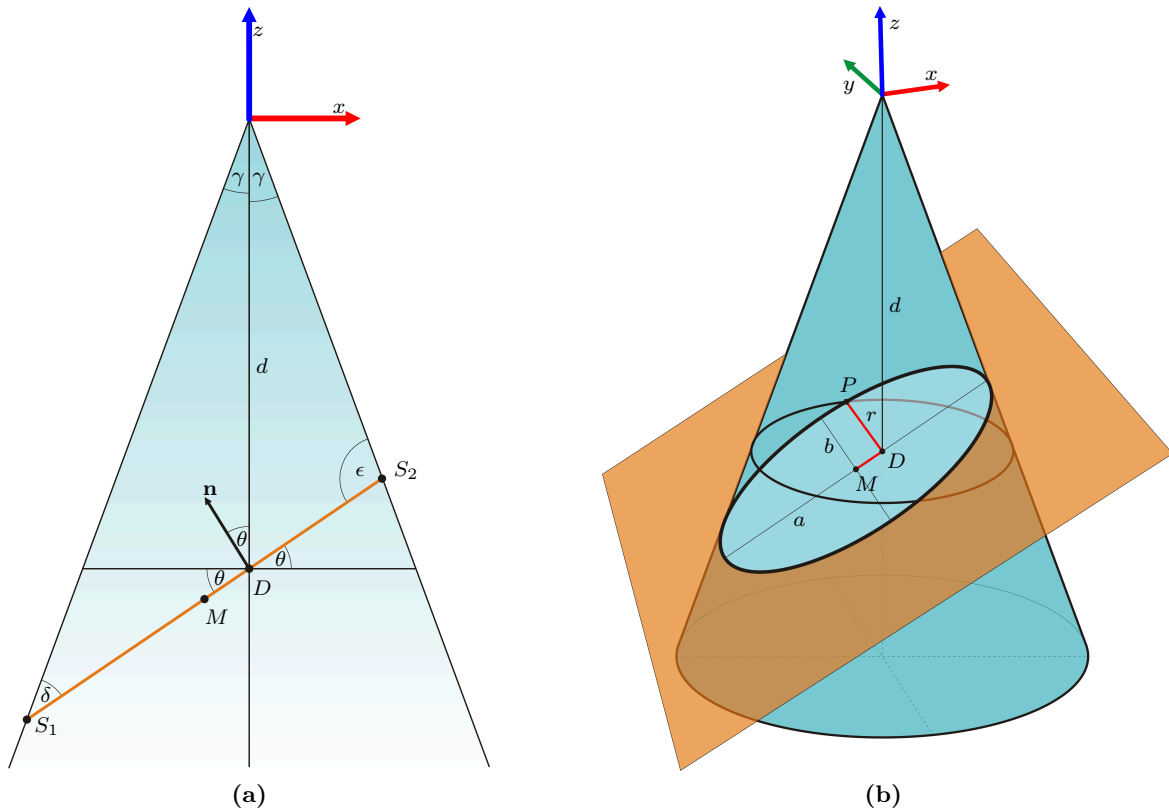


Abbildung A.1: Darstellung des von einer Ebene geschnittenen Kreiskegels als Seitenansicht der x-z-Ebene (a) und als dreidimensionale Ansicht (b).

und damit für die lange Halbachse

$$a = \frac{1}{2} (\overline{S_1 D} + \overline{D S_2}) = \frac{d}{2} \cdot \sin \gamma \cdot \left(\frac{1}{\cos(\theta - \gamma)} + \frac{1}{\cos(\theta + \gamma)} \right). \quad (\text{A.1})$$

Für die Bestimmung der kurzen Halbachse ist zu beachten, dass der Mittelpunkt der Ellipse in der Regel nicht auf der Kegelhachse liegt und die Halbachse den Mantelkreis mit Radius r um den Punkt D daher nicht schneidet. Aus Abbildung A.1b ist jedoch ersichtlich, dass der Punkt P sowohl auf diesem Mantelkreis als auch auf der Schnittellipse liegt. In dem von den Ellipsenachsen aufgespannten Koordinatensystem (\tilde{x}, \tilde{y}) ergibt sich die Halbachse b somit durch Umformen der Ellipsengleichung $(\tilde{x}^2/a^2) + (\tilde{y}^2/b^2) = 1$ zu

$$b = \frac{\tilde{y}}{\sqrt{1 - (\tilde{x}^2/a^2)}}$$

und Einsetzen von P . Mit $\tilde{x} = \overline{MD} = a - \overline{D S_2}$, $\tilde{y} = r = d \cdot \tan \gamma$ sowie a nach Gleichung (A.1) ergibt sich nach Vereinfachung

$$b = \frac{d \cdot \tan(\gamma)}{\sqrt{1 - \tan^2(\gamma) \tan^2(\theta)}}.$$

Der Flächeninhalt der Ellipse ist damit

$$A = \pi ab = \frac{\pi d^2 \sin^2(\gamma) \cos(\theta)}{\cos(\gamma - \theta) \cos(\gamma + \theta) \sqrt{1 - \tan^2(\gamma) \tan^2(\theta)}}. \quad (\text{A.2})$$

An diesem Ausdruck ist des Weiteren zu erkennen, dass das Verhältnis der Schnittflächen A_1 und A_2 zweier paralleler Ebenen der Abstände d_1 und d_2 unabhängig von ihrer Ausrichtung

$$\frac{A_1}{A_2} = \frac{d_1^2}{d_2^2} \quad (\text{A.3})$$

beträgt. Aufgrund des Strahlensatzes ist es dabei unerheblich, ob die Abstände d_1 und d_2 an der Kegelachse oder als kürzeste Abstände von der Kegelspitze senkrecht zu den Ebenen gemessen werden.



GLSL-Methoden zur Kubikwurzelberechnung

Da GLSL keinen Befehl für die Berechnung einer Kubikwurzel mit doppelter Genauigkeit bereitstellt, wurde im Rahmen der Simulation folgende Funktionen implementiert:

Vertex shader (Hilfsfunktionen zur Kubikwurzelberechnung)

```
1 // Schätzung der Kubikwurzel von float x
2 float magicCubeRoot(in float x)
3 {
4     int i = floatBitsToInt(x); // Reinterpretiere x als Integer
5     i = (0x2a517d3c + int(0.333333f * i)); // "Bit-Magie"
6     float y = intBitsToFloat(i); // Reinterpretiere als Float
7     return 0.666666f*y + 0.333333*x/(y*y); // Newton Iter. zur Verbesserung
8 }
9
10 // Kubikwurzelberechnung für double-Werte x
11 double newtonCubeRoot(in double x)
12 {
13     double v=sign(x)*magicCubeRoot(float(abs(x))); // Startwert
14     double v2 = (2*v*v*v + x) / (3*v*v); // Erste Iteration
15
16     int iter = 0;
17     while (abs(v2-v) > 1e-6 && iter < 200) // Weitere Iterationen
18     {
19         v = v2;
20         v2 = (2*v*v*v + x) / (3*v*v);
21         ++iter;
22     }
23
24     return v2;
25 }
```

Die Funktion `newtonCubeRoot` berechnet hier die Kubikwurzel des gegebenen Wertes `double x` als Nullstelle der Funktion $f(v) = v^3 - x$ mit dem Newton-Verfahren, also der Vorschrift

$$v_{n+1} = v_n - \frac{f(v_n)}{f'(v_n)} = v_n - \frac{v_n^3 - x}{3v_n^2} = \frac{2v_n^3 + x}{3v_n^2}.$$

Als Initialwert dient dabei eine Schätzung für die Kubikwurzel von `x` in einfacher Genauigkeit, welche von der Routine `magicCubeRoot` berechnet wird. Das Vorgehen basiert hier auf der Ausnutzung verschiedener Eigenschaften der Konvertierung von IEEE-konformen `float`-Werten in Integerdarstellung und erlaubt eine besonders effiziente und genaue Schätzung der Kubikwurzel. Der Urheber dieses „Hacks“ ist unbekannt, das Verfahren wurde jedoch mehrfach für die Berechnung inverser Kubikwurzeln ($1/\sqrt[3]{x}$) untersucht und bestätigt (beispielsweise von C. Lomont [93]). Die hier verwendete Variante für die normale, nicht invertierte Kubikwurzel geht auf C.P. Hansen zurück [67].



Shadercode zur Berechnung bahnbezogener Zielgrößen

Im Folgenden ist der Programmcode des Fragmentshaders zur Berechnung der bahnbezogenen Zielgrößen als Ergänzung zu den Beschreibungen des Algorithmus in Kapitel 5.5.2 gegeben. Der zugehörige Vertexshader realisiert lediglich die Standardoperation `gl_Position = ftransform()`.

Fragmentsheader zur Berechnung der bahnbezogenen Zielgrößen

```
1 #version 420 compatibility
2 #pragma optionNV(fastmath off)
3
4 // Technische Variablen
5 uniform int reduceOperation;
6 uniform int numBoundaryVertices;
7 uniform int numVertices;
8 uniform int reductionFBOsize;
9
10 // Faktoren zur Normalisierung
11 uniform vec4 postReductionFactors;
12
13 // Oberflächenzielgrößen im aktuellen und vorherigen Zeitschritt
14 uniform sampler2D currentSurfaceValues;
15 uniform sampler2D previousSurfaceValues;
16
17 // Bereits berechnete Momentanwerte
18 uniform sampler2D currentStepValues;
19
20 // Ergebnisse des vorherigen Reduktionsschritts
21 uniform sampler2D loopIn_Reduction1;
22 uniform sampler2D loopIn_Reduction2;
23 uniform sampler2D loopIn_Reduction3;
24
25 // Ergebnisausgabe der weiter reduzierten Werte
```

```

26 out vec4 Reduction1;
27 out vec4 Reduction2;
28 out vec4 Reduction3;
29
30 void main( void ) {
31
32     // Initialisierungsschritt
33     if (reduceOperation == 0)
34     {
35         // Berechne Messpunkt-Index
36         int index = int( trunc( gl_FragCoord.y ) * reductionFBOsize
37                         + trunc( gl_FragCoord.x ) );
38
39         // Lade oberflächenbezogene Werte
40         vec4 curValues
41             = texelFetch( currentSurfaceValues , ivec2( gl_FragCoord.xy ), 0);
42         vec4 prevValues
43             = texelFetch( previousSurfaceValues , ivec2( gl_FragCoord.xy ), 0);
44         vec4 curStep
45             = texelFetch( currentStepValues      , ivec2( gl_FragCoord.xy ), 0);
46
47         // Prüfe auf ungültige Rasterposition oder dV==0
48         if ( index >= numBoundaryVertices
49             || int( gl_FragCoord.x ) >= reductionFboSize
50             || ( ( curValues.g - prevValues.g ) < 0.5 ) )
51         {
52             // Setze neutrale Elemente
53             Reduction1 = vec4( 0 );
54             Reduction2 = vec4( -3.402823466e+38F );
55             Reduction3 = vec4( 3.402823466e+38F, 3.402823466e+38F, 0, 0 );
56         }
57         else
58         {
59             // Berechne Momentanwerte
60             Reduction1.r = curValues.r - prevValues.r;
61             Reduction1.g = Reduction1.r;
62             Reduction1.b = curValues.g - prevValues.g;
63
64             //  $F(v, t+1) * \Omega(v, t+1) - F(v, t) * \Omega(v, t)$ 
65             Reduction1.a = (curValues.a * curValues.r)
66                           - (prevValues.a * prevValues.r);
67             Reduction3.b = (curValues.b * curValues.r)
68                           - (prevValues.b * prevValues.r);
69
70             if (curStep.a >= 0.5) {
71                 // Innerhalb des Kernstrahls
72                 Reduction3.r = (curValues.b - prevValues.b)
73                               * (curValues.r / Reduction1.r) + prevValues.b;
74                 Reduction2.g = Reduction3.r;
75                 Reduction2.r = Reduction1.a / Reduction1.r;
76             } else {
77                 // außerhalb des Kernstrahls
78                 Reduction3.r = 3.402823466e+38F;
79                 Reduction2.g = -3.402823466e+38F;
80                 Reduction2.r = 0;
81             }
82         }
83     }
84 }

```

```

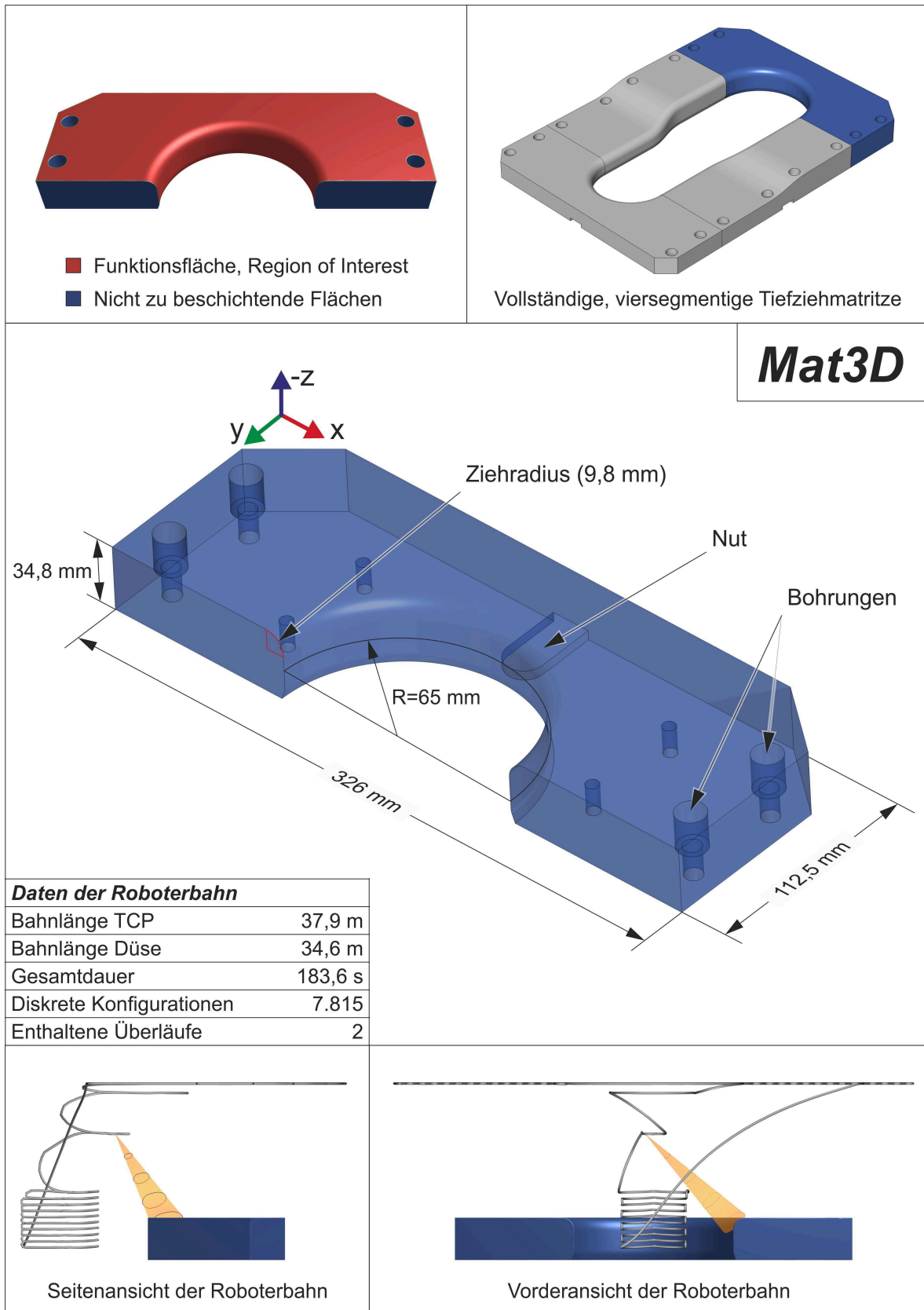
75
76 // Temperaturreduktion auch für nichtbeschichtete Vertices
77 if (index >= numBoundaryVertices || int(gl_FragCoord.x)>=reductionFboSize)
78 {
79     Reduction2.b = curStep.g;
80 }
81
82 // Temperatur auch für innere Vertices
83 if (index < numVertices) {
84     Reduction2.a = curStep.r;
85     Reduction3.ga = vec2(curStep.r);
86 }
87 }
88 else //reduceOperation == 1|2
89 {
90     ivec2 corner = ivec2(2.0*(gl_FragCoord.xy-0.5) + 0.5);
91
92     //Reduction1
93     vec4 val1 = texelFetch( loopIn_Reduction1, corner, 0 );
94     vec4 val2 = texelFetch( loopIn_Reduction1, corner+ivec2(1,0), 0 );
95     vec4 val3 = texelFetch( loopIn_Reduction1, corner+ivec2(0,1), 0 );
96     vec4 val4 = texelFetch( loopIn_Reduction1, corner+ivec2(1,1), 0 );
97     Reduction1 = val1 + val2 + val3 + val4;
98
99     //Reduction2
100    val1 = texelFetch( loopIn_Reduction2, corner, 0 );
101    val2 = texelFetch( loopIn_Reduction2, corner+ivec2(1,0), 0 );
102    val3 = texelFetch( loopIn_Reduction2, corner+ivec2(0,1), 0 );
103    val4 = texelFetch( loopIn_Reduction2, corner+ivec2(1,1), 0 );
104    Reduction2 = max( max( val1, val2 ), max( val3, val4 ) );
105
106    //Reduction3
107    val1 = texelFetch( loopIn_Reduction3, corner, 0 );
108    val2 = texelFetch( loopIn_Reduction3, corner+ivec2(1,0), 0 );
109    val3 = texelFetch( loopIn_Reduction3, corner+ivec2(0,1), 0 );
110    val4 = texelFetch( loopIn_Reduction3, corner+ivec2(1,1), 0 );
111    Reduction3.rg = min( min( val1.rg, val2.rg ), min( val3.rg, val4.rg ) );
112    Reduction3.ba = val1.ba + val2.ba + val3.ba + val4.ba;
113
114    // Normalisierung / Gewichtung
115    if (reduceOperation == 2) {
116        Reduction1.g *= postReductionFactors.r; // Effizienz
117        Reduction1.a /= Reduction1.r; // durchschnittl. Winkel
118        Reduction3.b /= Reduction1.r; // durchschnittl. Abstand
119        Reduction3.a /= numVertices; // durchschnittl. Temperatur
120    }
121 }
122
123 }

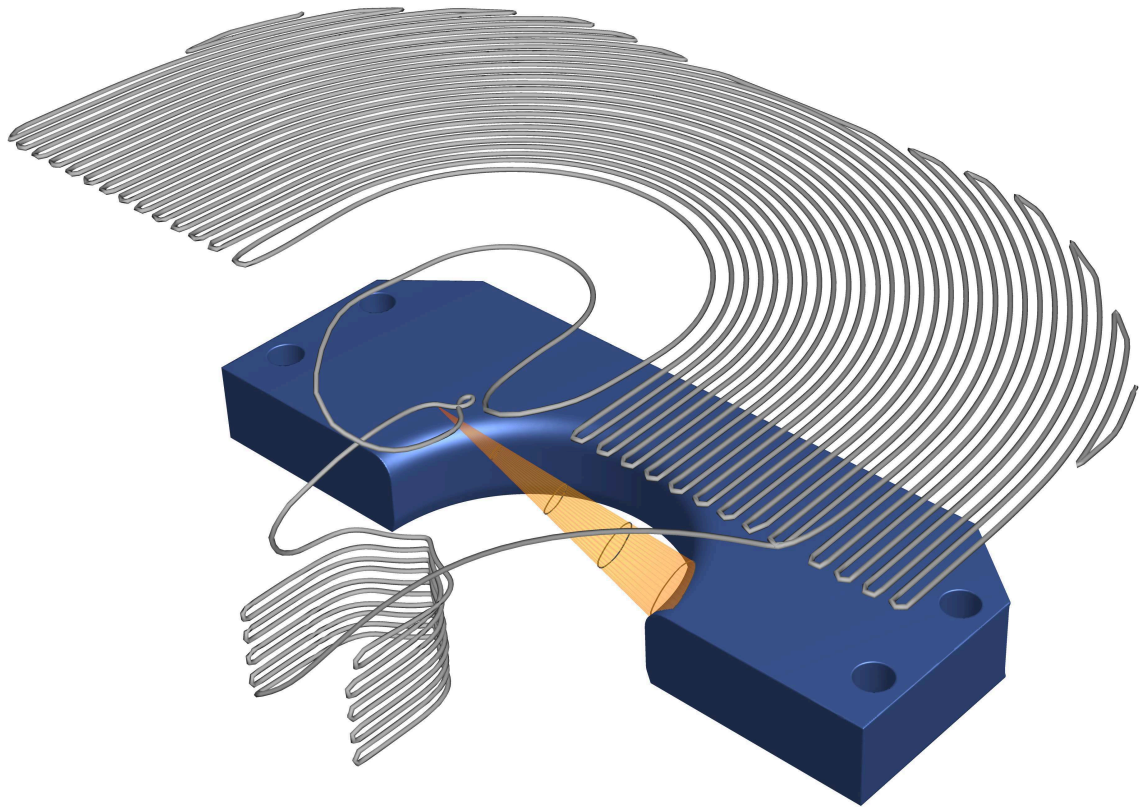
```


Beschichtete Matrizenbauteile

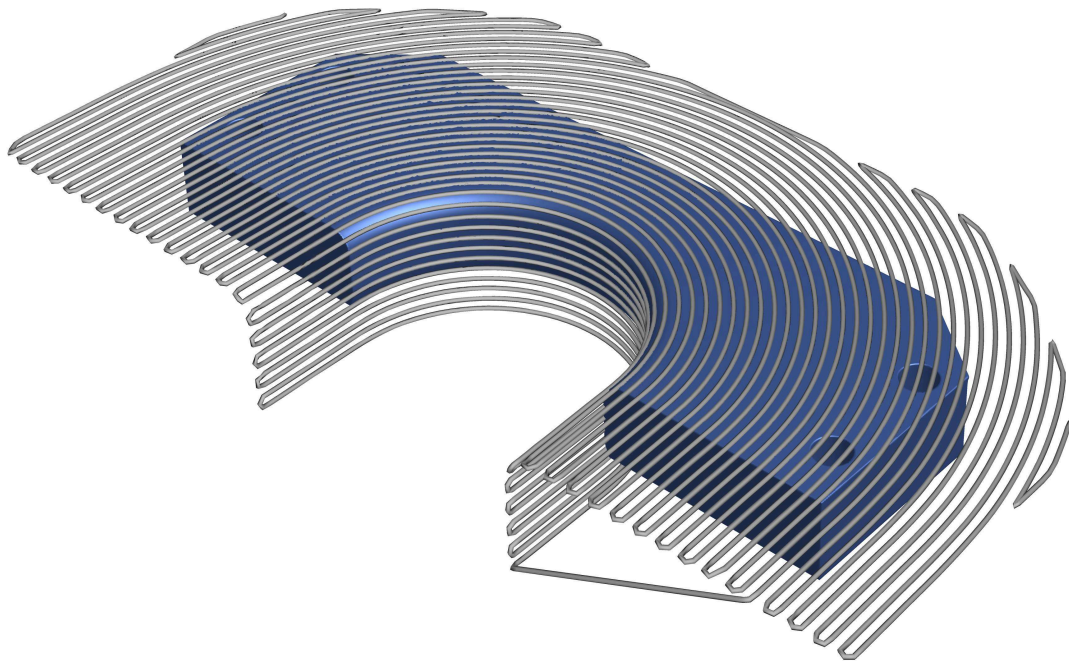
Auf den folgenden Seiten finden sich Darstellungen der Bauteile MAT3D und MAT2A, welche insbesondere an vielen Stellen in Kapitel 6 als Demonstrationsbauteile dienen. Die Abbildungen umfassen die Geometrie der Bauteile, die Darstellung der zu beschichtenden Funktionsflächen sowie verschiedene Ansichten und Kennwerte der zur Beschichtung verwendeten Roboterbahnen. Für das Bauteil MAT2A überdeckt die Bahn dabei absichtlich nicht die gesamte Funktionsfläche, um eine Auslaufzone der Schichtdicke auf dem Bauteil zu realisieren.

D. BESCHICHTETE MATRITZENBAUTEILE



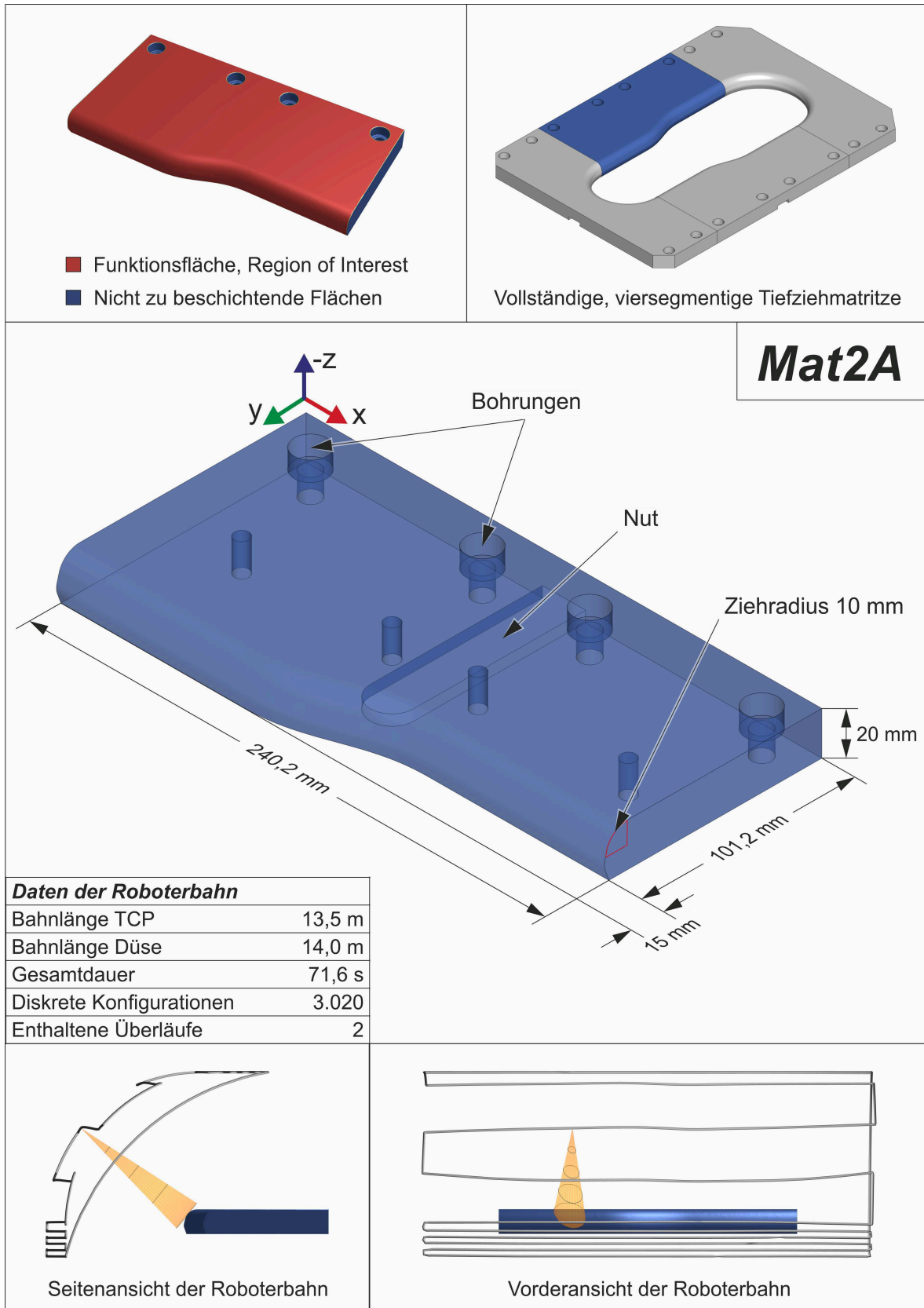


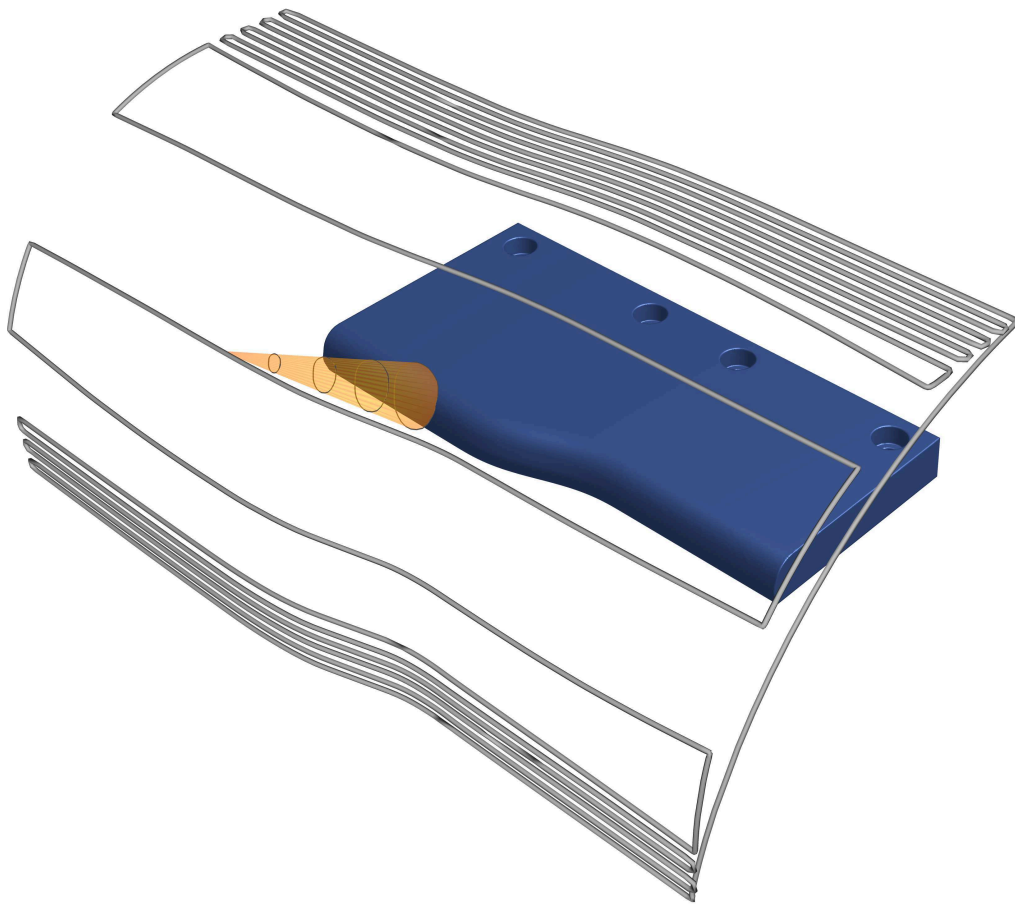
Roboterbahnverlauf an der Brennerdüse im Abstand von 110 mm zur Oberfläche



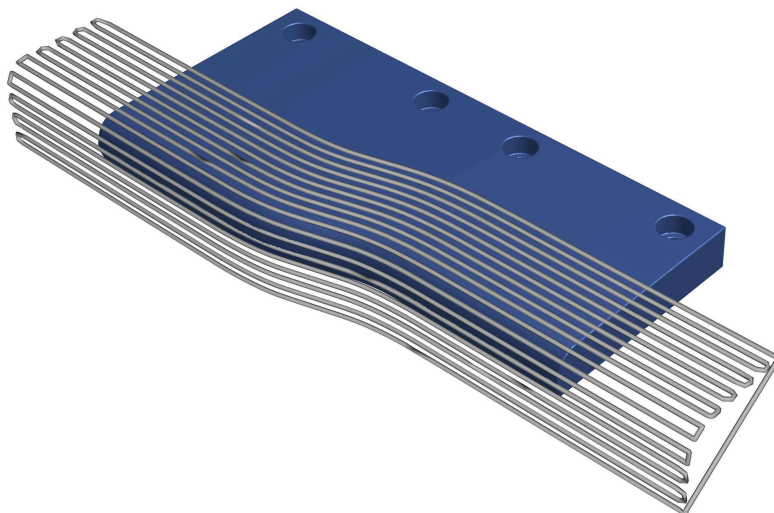
Roboterbahnverlauf am TCP mit geodätischem Bahnabstand von 5 mm

D. BESCHICHTETE MATRITZENBAUTEILE





Roboterbahnverlauf an der Brennerdüse im Abstand von 110 mm zur Oberfläche



Roboterbahnverlauf am TCP mit geodätischem Bahnabstand von 5 mm

Abbildungsverzeichnis

2.1	Verschiedene Splattformen	12
2.2	Illustration des mikrostrukturellen Schichtaufbaus	13
2.3	Lichtbogenspritzpistole (schematisch)	15
2.4	Tiefziehwerkzeug des SFB 708	17
2.5	Lichtbogenspritzpistole Smart Arc PPG (Foto)	18
2.6	REM Aufnahmen des Fülldrahts Durmat AS-850	19
3.1	Fertigung von Footprintprofilen	30
3.2	Datengewinnung durch Simulation oder Messung	34
3.3	Gesamtablauf der Footprintgewinnung	37
3.4	Abschirmung des Footprintsubstrats für das Sandstrahlen	39
3.5	Footprintproben (Fotos)	41
3.6	Koordinatensysteme	43
3.7	Varianten der Schlauchpaketführung	45
3.8	HVOF-Brenner (Foto)	47
3.9	Entwickelte Lasereinmessvorrichtung	49
3.10	Methodik zur Lasereinmessung	51
3.11	Aufgeschmolzene Beschichtung	52
3.12	3D-Scan eines thermisch verformten Footprints	53
3.13	Volumenmodell der Schichtmikrostruktur	57
3.14	Verwendete Scansysteme	59
3.15	Verwendetes Scanbahnraaster für Footprintproben	60
3.16	Vergleich mit verschiedenen Scansystemen erfasster Profile	61
3.17	3D-Scan eines Footprints	63
3.18	Ausrichtung des Scannerkoordinatensystems	64
3.19	Ausgerichteter Scandatensatz eines Footprints	65
3.20	Profilveränderung durch Sandstrahlen	67
3.21	Konturgrafiken verschiedener Fittingpolynome	70
3.22	Festgestellte thermische Verformungen	72
3.23	Eingebettete Footprint- und Linearbahnprobe	73
3.24	Querschnitte dreidimensionaler Footprintprofile	74
3.25	Genauigkeit des durchschnittlichen Footprints $\bar{\mathcal{F}}$	75

3.26	Segmentierungen unbeschichteter Footprintbereiche	76
3.27	Streuung gemessener Footprintprofile	77
3.28	Algorithmus zur Schichtdickenmessung	79
3.29	Ermittlung der Schichtmittelachse im Querschliffbild	81
3.30	Vergleich automatischer und manueller Schichtdickenmessungen	83
3.31	Abweichung der Simulation von gemessenen Linearbahnprofilen	85
3.32	Ergebnisse der Footprintoptimierung	89
3.33	Deformationsfunktionen und optimierte Profile	90
3.34	Originaler und optimierter Footprint	91
3.35	Optimierungsergebnis ohne Footprint als Startwert	91
3.36	Approximationsfehler in Abhängigkeit von α	92
3.37	Deformationsfunktionen in Abhängigkeit von α	93
3.38	Footprintgenauigkeit mit/ohne Ausgleich der thermischer Verformungen	94
3.39	Footprintprofile bei variierendem Spritzabstand	97
3.40	Abweichungen funktionsbasierter Footprintrepräsentationen	100
3.41	Konturen funktionsbasierter Footprintrepräsentationen	101
3.42	Footprints bei unterschiedlichen angenommenen Verfahrensgeschwindigkeiten	102
3.43	Funktionsbasierte Footprintrepräsentationen	103
3.44	3D-Spritzstrahlgeometrie	104
3.45	Spritzstrahlgeometrie (Seitenansichten)	105
3.46	Öffnungswinkel des Spritzstrahls	106
3.47	Abweichungen vom durchschnittlichen Footprint	108
4.1	Untersuchung von Turbulenzen vor dem Substrat	117
4.2	Geometrische Zusammenhänge im Beschichtungsmodell	119
4.3	Gültiger Parameterbereich der Übertragungsfunktion	122
4.4	Abstands- und Winkelabhängigkeit der geom. Übertragungsfunktion	123
4.5	Veranschaulichung der geometrischen Übertragungsfunktion	124
4.6	Abweichung der vereinfachten geometrischen Übertragungsfunktion	125
4.7	Fehler durch das approximative Beschichtungsmodell	126
4.8	Hermiteinterpolation	131
4.9	Beschichtung einer Flachprobe	133
4.10	Vor- und Nachlauf für Mäanderbahnen	133
4.11	Winkelabhängiger Verlauf des Haftwirkungsgrades	136
5.1	Grundlegender Ablauf der Beschichtungssimulation	138
5.2	Vernetzung der Bauteilgeometrie	146
5.3	Bahnplanungstool	147
5.4	Zweidimensionale Problemstruktur der Zielgrößenberechnung	151
5.5	Sichtbarkeitsberechnung und Footprint-Lookup	153
5.6	Übergeordneter Simulationsablauf und Datenfluss auf der GPU	157

5.7	Shaderpipeline zur Berechnung oberflächenbezogener Zielgrößen	159
5.8	Shaderpipeline zur Berechnung bahnbezogener Zielgrößen	172
5.9	Helixförmige Beschichtungsbahn	176
5.10	Zeitablauf eines Simulationsschritts	177
5.11	Simulationszeiten- und Zeitschrittdiagramme	179
6.1	Simulierter Footprintversuch	186
6.2	Numerische Probleme durch zu geringen Zeitschritt	187
6.3	Bahnzielgrößen am Beispiel geneigter Ebenen	190
6.4	Einfluss der Tiefenpufferauflösung auf den maximalen Schichtfehler	192
6.5	Einfluss der Tiefenpufferauflösung am Beispiel	193
6.6	Durchschnittlicher Schichtfehler abhängig von der Tiefenpufferauflösung	194
6.7	Simulationsgeschwindigkeit	198
6.8	Laufzeitvergleich mit Nvidia Optix	201
6.9	Beschichtung eines komplexen Bauteils (Fotos)	204
6.10	Schichtdickenmessung mit der Koordinatenmessmaschine Zeiss Prismo	205
6.11	Querschliffpositionen für komplexe Bauteile	207
6.12	Bauteilsegmentierung	208
6.13	Schichtdickenverläufe im Querschliff für das Bauteil MAT3D	209
6.14	Vergleich von Simulation und Messung auf komplexen Bauteilen	212
6.15	Krafteinwirkung des Schlauchpakets auf den Brenner	216
6.16	Simulierte sekundäre Oberflächenzielgrößen	219
6.17	Simulierte Bahnzielgrößen	220
6.18	Schichtdickenabweichungen unter Translation und Rotation der Bauteile	222
6.19	Ergebnisse der Sensitivitätsanalyse	223
6.20	Beispiel zur Bahnoptimierung	224
A.1	Ellipse als Kegelschnitt	232

Tabellenverzeichnis

2.1	Spritzwerkstoff Durmat AS-850	18
2.2	Standardkonfiguration für Lichtbogenspritzversuche	19
3.1	Vergleich verschiedener Scansysteme	56
3.2	Approximierbarkeit der Substratoberfläche durch Polynomfunktionen	68
5.1	Von der Simulation berechnete Zielgrößen	150
6.1	Hardwarekonfiguration des Testsystems	196
6.2	Vergleich zwischen taktilem Scan und Querschlifffmessung	208
6.3	Schichtrauheit und Messintervalle auf komplexen Bauteilen	210
6.4	Simulationsabweichung bei der Bauteilbeschichtung	214

Literaturverzeichnis

- [1] ABB Robotics: *Operating Manual Robotstudio 5.13, Rev. D*. 2010. URL: <http://developercenter.robotstudio.com/>.
- [2] A. Abedini, A. Pourmousa, S. Chandra und J. Mostaghimi: „Effect of substrate temperature on the properties of coatings and splats deposited by wire arc spraying“. In: *Surface and Coatings Technology* 201 (6 2006), S. 3350–3358. ISSN: 0257-8972. DOI: 10.1016/j.surfcoat.2006.07.184.
- [3] I. Ahmed und T. L. Bergman: „Thermal modeling of plasma spray deposition of nanostructured ceramics“. In: *Journal of Thermal Spray Technology* 8 (2 1999), S. 315–322. ISSN: 1059-9630. DOI: 10.1361/105996399770350539.
- [4] G. Antou und G. Montavon: „Quantifying Thermal Spray Coating Architecture by Stereological Protocols: Part II. Key Points to be Addressed“. In: *Journal of Thermal Spray Technology* 16 (2 2007), S. 168–176. ISSN: 1059-9630. DOI: 10.1007/s11666-007-9037-4.
- [5] T. Ba, C. Kang und H. Ng: „Numerical Study of the Plasma Flow Field and Particle In-flight Behavior with the Obstruction of a Curved Substrate“. In: *Journal of Thermal Spray Technology* 18 (5 2009), S. 858–874. ISSN: 1059-9630. DOI: 10.1007/s11666-009-9395-1.
- [6] F.-W. Bach, E. Lugscheider, T. Copitzky, J. Prehm u. a.: „Diagnostics at Thermal Coating Processes – Research Results of the DFG-Funded Project“. In: *Thermal Spray 2003: Advancing the science and applying the technology*. International Thermal Spray Conference. (Orlando, FL, USA, 5.–8. Mai 2003). Hrsg. von B. R. Marple und C. Moreau. Materials Park OH: ASM International, 2003, S. 1221–1227. ISBN: 0871707853.
- [7] F.-W. Bach, K. Möhwald, B. Dröbner, J. Prehm u. a.: „Influences on the Kinematics of the APS-Process Investigated by Means of Particle Image Velocimetry“. In: *Thermal Spray 2003: Advancing the science and applying the technology*. International Thermal Spray Conference. (Orlando, FL, USA, 5.–8. Mai 2003). Hrsg. von B. R. Marple und C. Moreau. Materials Park OH: ASM International, 2003, S. 1191–1196. ISBN: 0871707853.
- [8] F.-W. Bach, K. Möhwald, A. Laarmann und T. Wenz, Hrsg.: *Moderne Beschichtungsverfahren*. Weinheim: Wiley-VCH, 2005. 353 S. ISBN: 3-527-30977-2.

- [9] M. Berg, O. Cheong und M. van Kreveld: *Computational Geometry: Algorithms and Applications*. Berlin, Heidelberg: Springer, 2008. 386 S. ISBN: 978-3-54-077973-5.
- [10] R. Berthelsen, T. Wiederkehr, R. Denzer, A. Menzel und H. Müller: „Efficient Simulation of Nonlinear Heat Transfer during Thermal Spraying of Complex Workpieces“. In: *World Journal of Mechanics* 4 (9 2014), S. 289–301. ISSN: 2160-049X. DOI: 10.4236/wjm.2014.49029.
- [11] K. Bobzin: *Oberflächentechnik für den Maschinenbau*. Weinheim: Wiley-VCH, 2013. 439 S. ISBN: 978-3-527-33018-8.
- [12] P. Böckh und C. Saumweber: *Fluidmechanik. Einführendes Lehrbuch*. 3. Auflage. SpringerLink : Bücher. Wiesbaden: Springer Vieweg, 2013. 382 S. ISBN: 978-3-642-33892-5.
- [13] C.-A. Bohn: „Kohonen Feature Mapping through Graphics Hardware“. In: *Proceedings of International Conference on Computational Intelligence and Neurosciences*. (Research Triangle Park, NC, USA). 1998, S. 64–67.
- [14] V. Boljanovic: *Sheet Metal Forming Processes and Die Design*. New York: Industrial Press, 2004. 219 S. ISBN: 978-0-8311-3492-1.
- [15] R. Bolot, S. Deng, Z. Cai, H. Liao und G. Montavon: „A Coupled Model Between Robot Trajectories and Thermal History of the Workpiece During Thermal Spray Operation“. In: *Journal of Thermal Spray Technology* 23 (3 2013), S. 296–303. ISSN: 1059-9630. DOI: 10.1007/s11666-013-0048-z.
- [16] I. N. Bronstein, K. A. Semendjajew, G. Musiol und H. Mühlig: *Taschenbuch der Mathematik*. Frankfurt am Main: Verlag Harri Deutsch, 2000. 1234 S. ISBN: 978-3-81-712005-5.
- [17] R. W. Bulterman, F. W. van der Sommen, G. Zwaan, T. Verhoeff u. a.: „On computing a longest path in a tree“. In: *Information Processing Letters* 81 (2 2002), S. 93–96. ISSN: 0020-0190. DOI: 10.1016/S0020-0190(01)00198-3.
- [18] Z. Cai, S. Deng, H. Liao, C. Zeng und G. Montavon: „The Effect of Spray Distance and Scanning Step on the Coating Thickness Uniformity in Cold Spray Process“. In: *Journal of Thermal Spray Technology* 23 (3 2014), S. 354–362. ISSN: 1059-9630. DOI: 10.1007/s11666-013-0002-0.
- [19] F. C. Campbell, Hrsg.: *Metals fabrication. Understanding the basics*. ASM International, 2013. 449 S. ISBN: 978-1-62-708018-7.
- [20] A. Candel und R. Gadow: „Trajectory Generation and Coupled Numerical Simulation for Thermal Spraying Applications on Complex Geometries“. In: *Journal of Thermal Spray Technology* 18 (5 2009), S. 981–987. ISSN: 1059-9630.
- [21] J. Canny: „A Computational Approach to Edge Detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8 (6 1986), S. 679–698. ISSN: 0162-8828. DOI: 10.1109/TPAMI.1986.4767851.

- [22] E. Catmull und R. Rom: „A Class of Local Interpolating Splines“. In: *Computer aided geometric design. Proceedings of a conference held at the University of Utah, March 18-21, 1974*. Hrsg. von R. E. Barnhill und R. F. Riesenfeld. New York: Academic Press, 1974, S. 317–326. ISBN: 0120790505.
- [23] A. Chandak, I. Nagpur und G. Balasubramanian: „Reengineering of boiler feed pump using high performance coatings“. In: *Thermal Spray 2005: Proceedings of the International Thermal Spray Conference and Exposition*. (Basel, Schweiz, 2.–4. Mai 2005). Düsseldorf: DVS Media, 2005, S. 86–89.
- [24] B. Chapman, G. Jost und R. van der Pas: *Using OpenMP. Portable shared memory parallel programming*. Scientific and engineering computation. Cambridge, Massachusetts: MIT Press, 2008. 353 S. ISBN: 978-0-26-253302-7.
- [25] N. Chawla, V. V. Ganesh und B. Wunsch: „Three-dimensional (3D) microstructure visualization and finite element modeling of the mechanical behavior of SiC particle reinforced aluminum composites“. In: *Scripta Materialia* 51 (2 2004), S. 161–165. ISSN: 1359-6462. DOI: 10.1016/j.scriptamat.2004.03.043.
- [26] Y. Chen, G. Wang und H. Zhang: „Numerical simulation of coating growth and pore formation in rapid plasma spray tooling“. In: *Proceedings of the 5th Asia-Pacific Conference on Plasma Science & Technology and the 13th Symposium on Plasma Science for Materials* 390 (1–2 2001), S. 13–19. ISSN: 0040-6090. DOI: 10.1016/S0040-6090(01)00933-6.
- [27] Y. Chen, X. Liang, S. Wei, X. Chen und B. Xu: „Numerical Simulation of the Twin-Wire Arc Spraying Process: Modeling the High Velocity Gas Flow Field Distribution and Droplets Transport“. In: *Journal of Thermal Spray Technology* 21 (2 2012), S. 263–274. ISSN: 1059-9630. DOI: 10.1007/s11666-011-9723-0.
- [28] S. Cirolini, J. H. Harding und G. Jacucci: „Computer simulation of plasma-sprayed coatings I. Coating deposition model“. In: *Surface and Coatings Technology* 48 (2 1991), S. 137–145. ISSN: 0257-8972. DOI: 10.1016/0257-8972(91)90137-L.
- [29] T. W. Clyne und S. C. Gill: „Residual Stresses in Thermal Spray Coatings and Their Effect on Interfacial Adhesion: A Review of Recent Work“. In: *Journal of Thermal Spray Technology* 5 (4 1996), S. 401–418. ISSN: 1059-9630. DOI: 10.1007/BF02645271.
- [30] M. F. Cohen und J. R. Wallace: *Radiosity and realistic image synthesis*. Boston: Academic Press Professional, 1993. 381 S. ISBN: 978-0-12-178270-2.
- [31] R. L. Cook und K. E. Torrance: „A reflectance model for computer graphics“. In: *SIGGRAPH Computer Graphics* 15 (3 1981), S. 307–316. DOI: 10.1145/965161.806819.
- [32] P. I. Corke: *Robotics, vision and control. Fundamental algorithms in MATLAB*. Bd. 73. Springer tracts in advanced robotics. Berlin, Heidelberg: Springer, 2011. 570 S. ISBN: 978-3-64-220143-1.

- [33] C. Cui, U. Fritsching und A. Schulz: „Three-Dimensional Mathematical Modeling and Numerical Simulation of Billet Shape in Spray Forming Using a Scanning Gas Atomizer“. In: *Metallurgical and Materials Transactions B* 38 (2 2007), S. 333–346. ISSN: 1073-5615. DOI: 10.1007/s11663-007-9036-3.
- [34] C. Cui, U. Fritsching, A. Schulz und Q. Li: „Mathematical modeling of spray forming process of tubular preforms“. In: *Acta Materialia* 53 (9 2005), S. 2765–2774. ISSN: 1359-6454. DOI: 10.1016/j.actamat.2005.02.047.
- [35] S. Dallaire, M. Dufour und B. Gauthier: „Characterization of wear damage in coatings by optical profilometry“. In: *Journal of Thermal Spray Technology* 2 (4 1993), S. 363–368. ISSN: 1059-9630. DOI: 10.1007/BF02645866.
- [36] H. Dammertz und A. Keller: „Improving Ray Tracing Precision by Object Space Intersection Computation“. In: *2006 IEEE Symposium on Interactive Ray Tracing*. (Salt Lake City, UT, USA). 2006, S. 25–31. DOI: 10.1109/RT.2006.280211.
- [37] R. Danzl, F. Helmlí und S. Scherer: „Focus Variation – A New Technology For High Resolution Optical 3D Surface Metrology“. In: *Proceedings of 10th International Conference of the Slovenian Society for Non-Destructive Testing. Application of Contemporary Non-Destructive Testing in Engineering*. (Ljubljana, Slowenien, 2009). 2009, S. 484–491.
- [38] J. R. Davis: *Handbook of thermal spray technology*. Materials Park, Ohio: ASM International, 2004. 338 S. ISBN: 978-0-87-170795-6.
- [39] J. Denavit und R. Hartenberg: „A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices“. In: *Journal of Applied Mechanics* 22 (1955), S. 215–221. ISSN: 0021-8936.
- [40] S. Deng, H. Liao, C. Zeng, P. Charles und C. Coddet: „Robotic trajectory autogeneration in thermal spraying“. In: *Thermal Spray 2005: Proceedings of the International Thermal Spray Conference and Exposition*. (Basel, Schweiz, 2.–4. Mai 2005). Düsseldorf: DVS Media, 2005, S. 481–485.
- [41] S. Deng, Z. Cai, D. Fang, H. Liao und G. Montavon: „Application of robot offline programming in thermal spraying“. In: *Surface and Coatings Technology* 206 (19-20 2012), S. 3875–3882. ISSN: 0257-8972. DOI: 10.1016/j.surfcoat.2012.03.038.
- [42] S. Deng, H. Li, H. Liao, C. Coddet u. a.: „New Functions of Thermal Spray Toolkit - A Software Developed for Off-Line and Rapid Robot Programming“. In: *Thermal Spray 2006: Building on 100 Years of Success*. Proceedings of the 2006 International Thermal Spray Conference. (Seattle, WA, USA, 15.–18. Mai 2006). Hrsg. von B. R. Marple, M. M. Hyland, Y.-C. Lau, R. S. Lima und J. Voyer. Materials Park OH: ASM International, 2006, S. 1437–1442. ISBN: 978-0-87170-852-6.

- [43] S. Deng, H. Liang, Z. Cai, H. Liao und G. Montavon: „Kinematic Optimization of Robot Trajectories for Thermal Spray Coating Application“. In: *Journal of Thermal Spray Technology* 23 (8 2014), S. 1382–1389. ISSN: 1059-9630. DOI: 10.1007/s11666-014-0137-7.
- [44] Z. Djurić: „Three-dimensional simulation of liquid metal spray deposition onto arbitrary surfaces“. In: *Modelling and Simulation in Materials Science and Engineering* 12 (3 2004), S. 529–544. ISSN: 0965-0393.
- [45] Z. Djurić und P. Grant: „An inverse problem in modelling liquid metal spraying“. In: *Applied Mathematical Modelling* 27 (5 2003), S. 379–396. ISSN: 0307904X. DOI: 10.1016/S0307-904X(03)00044-1.
- [46] Z. Djurić und P. Grant: „Two-dimensional simulation of liquid metal spray deposition onto a complex surface: II. Splashing and redeposition“. In: *Modelling and Simulation in Materials Science and Engineering* 9 (2 2001), S. 111–127. ISSN: 0965-0393.
- [47] Z. Djurić, G. Newbery und P. Grant: „Two-dimensional simulation of liquid metal spray deposition onto a complex surface“. In: *Modelling and Simulation in Materials Science and Engineering* 7 (4 1999), S. 553–571. ISSN: 0965-0393.
- [48] S. Duncan, P. Jones und P. Wellstead: „A frequency-domain approach to determining the path separation for spray coating“. In: *IEEE Transactions on Automation Science and Engineering* 2 (3 2005), S. 233–239. ISSN: 1545-5955. DOI: 10.1109/TASE.2005.850393.
- [49] J. N. England: „A system for interactive modeling of physical curved surface objects“. In: *SIGGRAPH '78: Proceedings of the 5th annual conference on Computer graphics and interactive techniques*. (Atlanta, GA, USA, 1978). Hrsg. von S. H. Chasen und R. L. Phillips. New York, NY: ACM, 1978, S. 336–340. DOI: 10.1145/800248.807412.
- [50] W. R. Esposito und C. A. Floudas: „Gauss–Newton Method: Least Squares, Relation to Newton’s Method“. In: *Encyclopedia of Optimization*. Hrsg. von C. A. Floudas und P. M. Pardalos. New York, NY: Springer US, 2009, S. 1129–1134. ISBN: 978-0-387-74758-3.
- [51] A. Fahr, S. Giguere, B. Roge und K. McRae: „Application of NDE in aerospace coatings“. In: *Smart Nondestructive Evaluation for Health Monitoring of Structural and Biological Systems. SPIE Proceedings*. (San Diego, CA, 17. März 2002). Hrsg. von T. Kundu. 2002, S. 83–92. DOI: 10.1117/12.469867.
- [52] D. Fang, S. Deng, H. Liao und C. Coddet: „The Effect of Robot Kinematics on the Coating Thickness Uniformity“. In: *Journal of Thermal Spray Technology* 19 (4 2010), S. 796–804. ISSN: 1059-9630. DOI: 10.1007/s11666-010-9470-7.

- [53] D. Fang, S. Deng, H. Liao und C. Coddet: „The improvements of Thermal Spray Toolkit - extended software for robotic off-line programming“. In: *Thermal Spray 2010: Global Solutions for Future Application*. International Thermal Spray Conference & Exposition. (Singapur, 3. Mai 2010–5. Mai 2005). Hrsg. von DVS - German Welding Society. Düsseldorf: DVS Media, 2010, S. 522–526. ISBN: 978-3-87155-590-9.
- [54] M. M. Fasching, F. B. Prinz und L. E. Weiss: „Planning robotic trajectories for thermal spray shape deposition“. In: *Journal of Thermal Spray Technology* 2 (1 1993), S. 45–57. ISSN: 1059-9630. DOI: 10.1007/BF02647423.
- [55] P. Fauchais: „Understanding plasma spraying“. In: *Journal of Physics D: Applied Physics* 37 (9 2004), R86–R108. ISSN: 0022-3727. DOI: 10.1088/0022-3727/37/9/R02.
- [56] P. Fauchais, M. Fukumoto, A. Vardelle und M. Vardelle: „Knowledge Concerning Splat Formation: An Invited Review“. In: *Journal of Thermal Spray Technology* 13 (3 2004), S. 337–360. ISSN: 1059-9630. DOI: 10.1361/10599630419670.
- [57] P. Fauchais und M. Vardelle: „Sensors in Spray Processes“. In: *Journal of Thermal Spray Technology* 19 (4 2010), S. 668–694. ISSN: 1059-9630. DOI: 10.1007/s11666-010-9485-0.
- [58] Fritsch, F. N. und Carlson, R. E.: „Monotone Piecewise Cubic Interpolation“. In: *SIAM Journal on Numerical Analysis* 17 (2 1980), S. 238–246. ISSN: 00361429. DOI: 10.2307/2156610.
- [59] R. Gadow, A. Candel und M. Floristán: „Optimized robot trajectory generation for thermal spraying operations and high quality coatings on free-form surfaces“. In: *Surface and Coatings Technology* 205 (4 2010), S. 1074–1079. ISSN: 0257-8972. DOI: 10.1016/j.surfcoat.2010.08.121.
- [60] R. Garimella und B. Swartz: *Curvature Estimation for Unstructured Triangulations of Surfaces*. Hrsg. von Los Alamos National Laboratory. 2003.
- [61] M. Gevelber, C. Cui, B. Vattiat, Z. Fieldman u. a.: „Real Time Control for Plasma Spray: sensor issues, torch nonlinearities, and control of coating thickness“. In: *Thermal Spray 2005: Proceedings of the International Thermal Spray Conference and Exposition*. (Basel, Schweiz, 2.–4. Mai 2005). Düsseldorf: DVS Media, 2005, S. 667–672.
- [62] R. Ghafouri-Azar, J. Mostaghimi und S. Chandra: „Modeling development of residual stresses in thermal spray coatings“. In: *Computational Materials Science* 35 (1 2006), S. 13–26. ISSN: 09270256. DOI: 10.1016/j.commatsci.2005.02.007.
- [63] R. Ghafouri-Azar, J. Mostaghimi, S. Chandra und M. Charmchi: „A Stochastic Model to Simulate the Formation of a Thermal Spray Coating“. In: *Journal of Thermal Spray Technology* 12 (1 2003), S. 53–69. ISSN: 1059-9630.
- [64] R. C. Gonzalez und R. E. Woods: *Digital image processing*. 3rd Edition. Upper Saddle River, N.J.: Prentice Hall, 2008. 954 S. ISBN: 978-0-13-168728-8.

- [65] K.-H. Grote: *Dubbel. Taschenbuch für den Maschinenbau*. Hrsg. von J. Feldhusen. Berlin, Heidelberg: Springer, 2014. 2072 S. ISBN: 978-3-64-238890-3.
- [66] A. Hansbo und P. Nylén: „Models for the simulation of spray deposition and robot motion optimization in thermal spraying of rotating objects“. In: *Surface and Coatings Technology* 122 (2-3 1999), S. 191–201. ISSN: 0257-8972. DOI: 10.1016/S0257-8972(99)00255-8.
- [67] C. P. Hansen: *0x5f3759df (Titel)*. 2012. URL: <http://h14s.p5r.org/2012/09/0x5f3759df.html>.
- [68] J.-K. Hao, E. Lutton, E. Ronald, M. Schoenauer und D. Snyers, Hrsg.: *Lecture Notes in Computer Science. Artificial Evolution*. Bd. 1363. Berlin, Heidelberg: Springer, 1998. ISBN: 3-540-64169-6. DOI: 10.1007/BFb0026588.
- [69] J. H. Hattel und N. H. Pryds: „A unified spray forming model for the prediction of billet shape geometry“. In: *Acta Materialia* 52 (18 2004), S. 5275–5288. ISSN: 1359-6454. DOI: 10.1016/j.actamat.2004.07.016.
- [70] D. Hegels und H. Müller: „Evolutionary path generation for reduction of thermal variations in thermal spray coating“. In: *GECCO '13: Proceeding of the fifteenth annual conference on genetic and evolutionary computation*. (Amsterdam, Niederlande). Hrsg. von E. Alba und C. Blum. New York: ACM, 2013, S. 1277–1284. ISBN: 978-1-4503-1963-8. DOI: 10.1145/2463372.2463521.
- [71] D. Hegels, T. Wiederkehr und H. Müller: „Simulation based iterative post-optimization of paths of robot guided thermal spraying“. In: *Robotics and Computer-Integrated Manufacturing* 35 (2015), S. 1–15. ISSN: 0736-5845. DOI: 10.1016/j.rcim.2015.02.002.
- [72] K. E. Hoff, J. Keyser, M. Lin, D. Manocha und T. Culver: „Fast computation of generalized Voronoi diagrams using graphics hardware“. In: *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. SIGGRAPH '99. (Los Angeles, CA, USA, 8.–13. Aug. 1999). Hrsg. von W. Waggenspack. New York, NY, USA: Addison-Wesley Publishing Co., 1999, S. 277–286. DOI: 10.1145/311535.311567.
- [73] Š. Houdková, M. Kašparová und F. Zahálka: „The Influence of Spraying Angle on Properties of HVOF Sprayed Hardmetal Coatings“. In: *Journal of Thermal Spray Technology* 19 (5 2010), S. 893–901. ISSN: 1059-9630. DOI: 10.1007/s11666-010-9514-z.
- [74] J. J. Jiménez, C. J. Ogáyar, J. M. Noguera und F. Paulano: „Performance Analysis for GPU-based Ray-triangle Algorithms“. In: *Proceedings of the International Conference on Computer Graphics Theory and Applications (GRAPP 2014)*. International Conference on Computer Graphics Theory and Applications. (Lissabon, Portugal, 5.–9. Jan. 2014). 2014, S. 239–246. ISBN: 978-989-758-002-4.

- [75] H. Jungklaus: „Thermal Spray Coatings in Printing and Paper Making“. In: *Proceedings of ITSC International Thermal Spray Conference. 02. – 04. 06. 2008, Maastricht, Niederlande.* 2008, S. 195–199.
- [76] C. W. Kang, H. W. Ng und S. C. M. Yu: „Plasma Spray Deposition on Inclined Substrates: Simulations and Experiments“. In: *Journal of Thermal Spray Technology* 16 (2 2007), S. 261–274. ISSN: 1059-9630. DOI: 10.1007/s11666-007-9024-9.
- [77] O. Kesler, M. Finot, S. Suresh und S. Sampath: „Determination of processing-induced stresses and properties of layered and graded coatings: Experimental method and results for plasma-sprayed Ni-Al₂O₃“. In: *Acta Materialia* 45 (8 1997), S. 3123–3134. ISSN: 1359-6454. DOI: 10.1016/S1359-6454(97)00015-3.
- [78] J. Kessenich und LunarG: *The OpenGL Shading Language.* The Khronos Group Inc. 2014. URL: <https://www.opengl.org/registry/doc/GLSLangSpec.4.40.pdf>.
- [79] S. V. Klinkov und V. F. Kosarev: „Measurements of Cold Spray Deposition Efficiency“. In: *Journal of Thermal Spray Technology* 15 (3 2006), S. 364–371. ISSN: 1059-9630. DOI: 10.1361/105996306X124365.
- [80] J. König, M. Lahres und O. T. Methner: „Quality designed twin wire arc spraying of aluminum bores“. In: *Thermal Spray 2014: Proceedings of the International Thermal Spray Conference.* International Thermal Spray Conference & Exhibition. (Barcelona, Spanien, 21.–23. Mai 2014). Hrsg. von DVS - German Welding Society. Düsseldorf: DVS Media, 2014, S. 227–233. ISBN: 978-3-87155-574-9.
- [81] A. Kout und H. Müller: „Quantitative improvement of tool impact paths defined by isolines of scalar functions on triangular mesh workpiece surfaces“. In: *The International Journal of Advanced Manufacturing Technology* 70 (1-4 2014), S. 237–255. ISSN: 0268-3768. DOI: 10.1007/s00170-013-5152-1.
- [82] A. Kout und H. Müller: „Tool-adaptive offset paths on triangular mesh workpiece surfaces“. In: *Computer-Aided Design* 50 (2014), S. 61–73. ISSN: 00104485. DOI: 10.1016/j.cad.2014.01.009.
- [83] A. Kout, T. Wiederkehr und H. Müller: „Efficient stochastic simulation of thermal spray processes“. In: *Surface and Coatings Technology* 203 (12 2009), S. 1580–1595. ISSN: 0257-8972. DOI: 10.1016/j.surfcoat.2008.11.035.
- [84] M. Kraume: *Transportvorgänge in der Verfahrenstechnik. Grundlagen und apparative Umsetzungen.* 2. Aufl. 2012. Berlin, Heidelberg: Springer, 2012. 646 S. ISBN: 978-3-64-225149-8.
- [85] B. Krebs: „Konturgenaue Bauteilbeschichtung für den Verschleißschutz mittels atmosphärischen Plasmaspritzens und Lichtbogenspritzens“. LWT. Dissertation. Dortmund: TU Dortmund, 2011.

- [86] S. Kuroda und T. W. Clyne: „The quenching stress in thermally sprayed coatings“. In: *Thin Solid Films* 200 (1 1991), S. 49–66. ISSN: 0040-6090. DOI: 10.1016/0040-6090(91)90029-w.
- [87] S. Kuroda, T. Fukushima und S. Kitahara: „Generation mechanisms of residual stresses in plasma-sprayed coatings. Selected proceedings of the 11th international vacuum congress (IVC-11) 7th international conference on solid surfaces (ICSS-7)“. In: *Vacuum* 41 (4-6 1990), S. 1297–1299. ISSN: 0042-207X. DOI: 10.1016/0042-207X(90)93938-F.
- [88] K. Lange: *Umformtechnik*. Berlin, Heidelberg: Springer, 1990. 632 S. ISBN: 978-3-662-10687-7. DOI: 10.1007/978-3-662-10686-0.
- [89] S. H. Leigh und C. C. Berndt: „Evaluation of off-angle thermal spray“. In: *Surface and Coatings Technology* 89 (3 1997), S. 213–224. ISSN: 0257-8972. DOI: 10.1016/S0257-8972(96)02897-6.
- [90] J. Lengyel, M. Reichert, B. R. Donald und D. P. Greenberg: „Real-time robot motion planning using rasterizing computer graphics hardware“. In: *SIGGRAPH '90: Proceedings of the 17th annual conference on Computer graphics and interactive techniques*. (Dallas, TX, USA, 6.–10. Aug. 1990). Hrsg. von F. Baskett. New York, NY: ACM, 1990, S. 327–335. ISBN: 0-89791-344-2. DOI: 10.1145/97879.97915.
- [91] C.-J. Li, W.-Y. Li und H. Liao: „Examination of the Critical Velocity for Deposition of Particles in Cold Spraying“. In: *Journal of Thermal Spray Technology* 15 (2 2006), S. 212–222. ISSN: 1059-9630. DOI: 10.1361/105996306X108093.
- [92] Y. J. Lin, E. J. Lavernia, J. E. Bobrow und D. R. White: „Modeling of spray-formed materials: Geometrical considerations“. In: *Metallurgical and Materials Transactions A* 31 (11 2000), S. 2917–2929. ISSN: 1073-5623. DOI: 10.1007/BF02830347.
- [93] C. Lomont: *Fast Inverse Square Root*. 2003. URL: <http://www.lomont.org/Math/Papers/2003/InvSqrt.pdf>.
- [94] E. Lugscheider und F.-W. Bach, Hrsg.: *Handbuch der thermischen Spritztechnik. Technologien - Werkstoffe - Fertigung*. Düsseldorf: Verlag für Schweißen und Verwandte Verfahren (DVS), 2002. 221 S. ISBN: 3871551864.
- [95] E. Lugscheider, K. Bobzin und R. Nickel: „Application of multiscale modeling in the coating formation simulation of APS PYSZ TBCs“. In: *Journal of Thermal Spray Technology* 15 (4 2006), S. 537–544. ISSN: 1059-9630.
- [96] E. Lugscheider und R. Nickel: „Finite element simulation of a coating formation on a turbine blade during plasma spraying“. In: *Surface and Coatings Technology* 174-175 (2003), S. 475–481. ISSN: 0257-8972. DOI: 10.1016/S0257-8972(03)00331-1.

- [97] X. Ma und P. Ruggiero: „Ultra-smooth hardface coatings applied by advanced HVOF process“. In: *Thermal Spray 2014: Proceedings of the International Thermal Spray Conference*. International Thermal Spray Conference & Exhibition. (Barcelona, Spanien, 21.–23. Mai 2014). Hrsg. von DVS - German Welding Society. Düsseldorf: DVS Media, 2014, S. 285–290. ISBN: 978-3-87155-574-9.
- [98] S. Markus, C. Cui und U. Fritsching: „Analysis of deposit growth in spray forming with multiple atomizers“. In: *Materials Science and Engineering: A* 383 (1 2004), S. 166–174. ISSN: 09215093. DOI: 10.1016/j.msea.2004.02.046.
- [99] J. Matejcek: „In situ measurement of residual stresses and elastic moduli in thermal sprayed coatings Part 1: apparatus and analysis“. In: *Acta Materialia* 51 (3 2003), S. 863–872. ISSN: 1359-6454. DOI: 10.1016/S1359-6454(02)00478-0.
- [100] P. Mathur, S. Annavarapu, D. Apelian und A. Lawley: „Spray casting: an integral model for process understanding and control“. In: *Materials Science and Engineering: A* 142 (2 1991), S. 261–276. ISSN: 09215093. DOI: 10.1016/0921-5093(91)90665-A.
- [101] G. Mauer, R. Vaßen und D. Stöver: „Comparison and Applications of DPV-2000 and Accuraspray-g3 Diagnostic Systems“. In: *Journal of Thermal Spray Technology* 16 (3 2007), S. 414–424. ISSN: 1059-9630. DOI: 10.1007/s11666-007-9047-2.
- [102] D. Meschede: *Gerthsen Physik*. Berlin, Heidelberg: Springer, 2006. 1162 S. ISBN: 3-540-25421-8. DOI: 10.1007/3-540-29973-4.
- [103] J. Mi und P. S. Grant: „Modelling the shape and thermal dynamics of Ni superalloy rings during spray forming Part 1: Shape modelling – Droplet deposition, splashing and redeposition“. In: *Acta Materialia* 56 (7 2008), S. 1588–1596. ISSN: 1359-6454. DOI: 10.1016/j.actamat.2007.12.021.
- [104] P. E. Mix: *Introduction to Nondestructive Testing. A training guide*. 2nd Edition. Hoboken, N.J.: John Wiley & Sons, Inc, 2005. 712 S. ISBN: 978-0-471-42029-3.
- [105] V. Moya, C. Gonzalez, J. Roca, A. Fernandez und R. Espasa: „Shader Performance Analysis on a Modern GPU Architecture“. In: *38th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO'05)*. (Barcelona, Spain, 12.–16. Nov. 2005). 2005, S. 355–364. DOI: 10.1109/MICRO.2005.30.
- [106] H. Müller: „Using Graphics Algorithms as Subroutines in Collision Detection“. In: *Graphics and Robotics*. Hrsg. von W. Straßer und F. Wahl. Berlin, Heidelberg: Springer, 1995, S. 45–57. ISBN: 978-3-540-58358-5.
- [107] A. Nadeau, L. Pouliot, F. Nadeau, J. Blain u. a.: „Online Coating Thickness Monitoring: Conclusive Results from Early Industrial Implementations“. In: *Thermal Spray 2007: Global Coating Solutions*. Proceedings of the 2007 International Thermal Spray Conference. (Beijing, China, 14.–16. Mai 2007). Hrsg. von B. R. Marple, M. M. Hyland, Y.-C. Lau, C.-J. Li u. a. Materials Park OH: ASM International, 2007, S. 860–865. ISBN: 978-0-87170-855-7.

- [108] J. Norberto Pires, H. Chen, T. Fuhlbrigge und X. Li: „A review of CAD-based robot path planning for spray painting“. In: *Industrial Robot: An International Journal* 36 (1 2009), S. 45–50. ISSN: 0143-991X. DOI: 10.1108/01439910910924666.
- [109] *Thermisches Spritzen - Begriffe, Einteilung*. Norm. Deutsches Institut für Normung, Beuth Verlag, Berlin, Juni 2005, ICS: 01.040.25, 25.220.20. EN 657:2005.
- [110] *Thermisches Spritzen - Pulver - Zusammensetzung, Technische Lieferbedingungen*. Norm. Deutsches Institut für Normung, Beuth Verlag, Berlin, Februar 2005, ICS 25.220.20; 77.160. EN 1274:2004.
- [111] Y. Okajima, D. Kudo, N. Okaya, T. Torigoe u. a.: „Evolution of Thermal Barrier Coatings for Land Based Gas Turbines at MHI“. In: *Thermal Spray 2014: Proceedings of the International Thermal Spray Conference*. International Thermal Spray Conference & Exhibition. (Barcelona, Spanien, 21.–23. Mai 2014). Hrsg. von DVS - German Welding Society. Düsseldorf: DVS Media, 2014, S. 257–262. ISBN: 978-3-87155-574-9.
- [112] J. D. Owens, D. Luebke, N. Govindaraju, M. Harris u. a.: „A Survey of General-Purpose Computation on Graphics Hardware“. In: *Computer Graphics Forum* 26 (1 2007), S. 80–113. ISSN: 0167-7055. DOI: 10.1111/j.1467-8659.2007.01012.x.
- [113] M. Pasandideh-Fard, R. Bhola, S. Chandra und J. Mostaghimi: „Deposition of tin droplets on a steel plate: simulations and experiments“. In: *International Journal of Heat and Mass Transfer* 41 (19 1998), S. 2929–2945. ISSN: 0017-9310. DOI: 10.1016/S0017-9310(98)00023-4.
- [114] M. Pasandideh-Fard, V. Pershin, S. Chandra und J. Mostaghimi: „Splat Shapes in a Thermal Spray Coating Process: Simulations and Experiments“. In: *Journal of Thermal Spray Technology* 11 (2 2002), S. 206–217. ISSN: 1059-9630. DOI: 10.1361/105996302770348862.
- [115] L. Pawłowski: *The science and engineering of thermal spray coatings*. 2nd Edition. Chichester, England und Hoboken, N.J.: Wiley, 2008. 656 S. ISBN: 978-0-471-49049-4.
- [116] N. H. Pryds, J. H. Hattel, T. B. Pedersen und J. Thorborg: „An integrated numerical model of the spray forming process“. In: *Acta Materialia* 50 (16 2002), S. 4075–4091. ISSN: 1359-6454. DOI: 10.1016/S1359-6454(02)00205-7.
- [117] S. Rausch, D. Biermann und P. Kersting: „Five-axis grinding of wear-resistant, thermally sprayed coatings on free-formed surfaces“. In: *Production Engineering* 8 (4 2014), S. 423–429. ISSN: 0944-6524. DOI: 10.1007/s11740-014-0537-z.
- [118] C. Reinsch: „Smoothing by spline functions“. In: *Numerische Mathematik* 10 (3 1967), S. 177–183. ISSN: 0029-599X.
- [119] K. Remesh, H. W. Ng und S. C. M. Yu: „Influence of Process Parameters on the Deposition Footprint in Plasma-Spray Coating“. In: *Journal of Thermal Spray Technology* 12 (3 2003), S. 377–392. ISSN: 1059-9630. DOI: 10.1361/105996303770348258.

- [120] K. Ruusuvoori, K. Lahdenperä, M. Oksa, E. Turunen und R. Lundberg: „Controlled removal of HVOF hard coatings by abrasive water jet“. In: *Thermal Spray 2005: Proceedings of the International Thermal Spray Conference and Exposition*. (Basel, Schweiz, 2.–4. Mai 2005). Düsseldorf: DVS Media, 2005, S. 1016–1020.
- [121] A. Sadovoy: „Modeling and offline simulation of thermal spray coating process for gas turbine applications“. Dissertation. Darmstadt: Technischen Universität Darmstadt, 2014. 109 S.
- [122] A. Sadovoy, R. Subramanian, S. Kycheck und A. Borchardt: „Optimization of thermal spray process development by off-line programming with coating thickness simulation“. In: *Proceedings of the 10th International Conference The "A" Coatings 2013*. The "A" Coatings. (Aachen, 10.–11. Apr. 2013). Hrsg. von K. Bobzin. Hrsg. von K.-D. Bouzakis, B. Denkena und M. Merklein. Bd. 27. Schriftenreihe Oberflächentechnik 27. Aachen: Shaker Verlag, 2013, S. 171–182. ISBN: 978-3-8440-1781-6.
- [123] D. Salomon: *Curves and surfaces for computer graphics*. New York: Springer, 2006. 460 S. ISBN: 978-0-387-24196-8.
- [124] G. Sansoni, S. Carmignato und E. Savio: „Validation of the measurement performance of a three dimensional vision sensor by means of a Coordinate Measurement Machine“. In: *Proceedings of IMTC 2004*. Como und Italien, 2004, S. 773–778.
- [125] J. L. Santos und F. Farahi, Hrsg.: *Handbook of optical sensors*. CRC Press, 2014. 718 S. ISBN: 978-1439866856.
- [126] E. Savio, L. Dechiffre und R. Schmitt: „Metrology of freeform shaped parts“. In: *CIRP Annals - Manufacturing Technology* 56 (2 2007), S. 810–835. ISSN: 0007-8506. DOI: 10.1016/j.cirp.2007.10.008.
- [127] D. Schaefers: „Ein Software-Tool zur Manipulation von Industrieroboterbahnen für extrudierte Werkstücke“. Lehrstuhl Informatik VII - Graphische Systeme. Bachelorarbeit. Dortmund: TU Dortmund, 2012.
- [128] K. E. Schneider: *Thermal spraying for power generation components*. Weinheim: Wiley-VCH, 2006. 271 S. ISBN: 3-527-31337-0.
- [129] H. Schwenke, U. Neuschaefer-Rube, T. Pfeifer und H. Kunzmann: „Optical Methods for Dimensional Metrology in Production Engineering“. In: *CIRP Annals - Manufacturing Technology* 51 (2 2002), S. 685–699. ISSN: 0007-8506. DOI: 10.1016/S0007-8506(07)61707-7.
- [130] M. Segal und K. Akeley: *The OpenGL Graphics System: A Specification. Version 4.5 core profile*. Hrsg. von The Khronos Group Inc. 2015. URL: <https://www.khronos.org/registry/doc/glspec45.core.pdf>.

- [131] J.-L. Seichepine, S. Albertazzi, P. Nardin, C. Coddet und D. Sporer: „Topological characterization of wear tracks in thermally sprayed abradable coatings“. In: *Thermal Spray 2005: Proceedings of the International Thermal Spray Conference and Exposition*. (Basel, Schweiz, 2.–4. Mai 2005). Düsseldorf: DVS Media, 2005, S. 1480–1484.
- [132] H.-K. Seok, K. Oh, H. Ra, D.-H. Yeo und H.-I. Lee: „A three-dimensional model of the spray forming method“. In: *Metallurgical and Materials Transactions B* 29 (3 1998), S. 699–708. ISSN: 1073-5615. DOI: 10.1007/s11663-998-0105-z.
- [133] J. R. Shewchuk: „What is a Good Linear Element? - Interpolation, Conditioning and Quality Measures“. In: *11th International Meshing Roundtable*. 2002, S. 115–126.
- [134] J. R. Shewchuk: *An introduction to the conjugate gradient method without the agonizing pain*. 1994. URL: <http://www.cs.berkeley.edu/~jrs/jrspapers.html#cg>.
- [135] F. Y. Shih: *Image Processing and Pattern Recognition*. Hoboken, N.J., USA: John Wiley & Sons, Inc, 2010. 326 S. ISBN: 978-0-470-59041-6. DOI: 10.1002/9780470590416.
- [136] K. Shoemake: „Animating rotation with quaternion curves“. In: *SIGGRAPH Computer Graphics* 19 (3 1985), S. 245–254. DOI: 10.1145/325165.325242.
- [137] H. Si: „Three Dimensional Boundary Conforming Delaunay Mesh Generation. Drei-dimensionale randkonforme Delaunay-Gitter Generierung“. Fakultät für Mathematik und Naturwissenschaften. Dissertation. Berlin: TU Berlin, 2008.
- [138] S. Siegmann und C. Abert: „100 years of thermal spray: About the inventor Max Ulrich Schoop“. In: *Surface and Coatings Technology* 220 (2013), S. 3–13. ISSN: 0257-8972. DOI: 10.1016/j.surfcoat.2012.10.034.
- [139] J. Spowart: „Automated serial sectioning for 3-D analysis of microstructures“. In: *Scripta Materialia* 55 (1 2006), S. 5–10. ISSN: 1359-6462. DOI: 10.1016/j.scriptamat.2006.01.019.
- [140] V. Srinivasan, M. Friis, A. Vaidya, T. Streibl und S. Sampath: „Particle Injection in Direct Current Air Plasma Spray: Salient Observations and Optimization Strategies“. In: *Plasma Chemistry and Plasma Processing* 27 (5 2007), S. 609–623. DOI: 10.1007/s11090-007-9089-8.
- [141] W. Tillmann, Hrsg.: *SFB 708 - 1. öffentliches Kolloquium*. Bd. 1. 3D-Surface Engineering für Werkzeugsysteme der Blechformteilefertigung. Dortmund: Praxiswissen, 2007. 181 S. ISBN: 978-3-89957-062-5.
- [142] W. Tillmann, Hrsg.: *SFB 708 - 2. öffentliches Kolloquium*. Bd. 2. 3D-Surface Engineering für Werkzeugsysteme der Blechformteilefertigung. Dortmund: Praxiswissen, 2008. 181 S. ISBN: 978-3-89957-072-4.
- [143] W. Tillmann, Hrsg.: *SFB 708 - 3. öffentliches Kolloquium*. Bd. 3. 3D-Surface Engineering für Werkzeugsysteme der Blechformteilefertigung. Dortmund: Praxiswissen, 2009. 181 S. ISBN: 978-3-86975-010-1.

- [144] W. Tillmann und I. Baumann, Hrsg.: *SFB 708 - 6. öffentliches Kolloquium*. Bd. 6. 3D-Surface Engineering für Werkzeugsysteme der Blechformteilefertigung. Dortmund: Praxiswissen, 2013. 222 S. ISBN: 978-3869750712.
- [145] W. Tillmann, I. Baumann, P. Hollingsworth und I.-A. Laemmerhirt: „Influence of the Spray Angle on the Properties of HVOF Sprayed WC–Co Coatings Using (–10 + 2µm) Fine Powders“. In: *Journal of Thermal Spray Technology* 22 (2-3 2013), S. 272–279. ISSN: 1059-9630. DOI: 10.1007/s11666-013-9882-2.
- [146] W. Tillmann, L. Hagen, M. Abdulgader, H.-G. Rademacher u. a.: „Investigations of Technological Developments in Nozzle Design for Twin Wire Arc Spraying by Means of Spray Plume Characteristics“. In: *Thermal Spray 2014: Proceedings of the International Thermal Spray Conference*. International Thermal Spray Conference & Exhibition. (Barcelona, Spanien, 21.–23. Mai 2014). Hrsg. von DVS - German Welding Society. Düsseldorf: DVS Media, 2014, S. 455–460. ISBN: 978-3-87155-574-9.
- [147] W. Tillmann, P. Hollingsworth, I. Baumann, S. Flossbach u. a.: „Influence of the Substrate’s Shape Complexity on the Coating Properties Produced by HVOF Spraying of Fine WC-12Co (2-10 µm) Powders“. In: *Thermal Spray 2013: Proceedings of the International Thermal Spray Conference*. International Thermal Spray Conference and Exposition. (Busan, South Korea, 13. Mai 2013–15. Mai 2005). Hrsg. von R. S. Lima, A. Agarwal, M. M. Hyland, Y.-C. Lau u. a. Materials Park OH: ASM International, 2013, S. 707–711.
- [148] W. Tillmann, P. Hollingsworth, I. Baumann, D. Hegels u. a.: „Adaption of the Path Strategy to Produce Near Net Shape Coatings Manufactured by HVOF Spraying“. In: *Thermal Spray 2014: Proceedings of the International Thermal Spray Conference*. International Thermal Spray Conference & Exhibition. (Barcelona, Spanien, 21.–23. Mai 2014). Hrsg. von DVS - German Welding Society. Düsseldorf: DVS Media, 2014, S. 950–955. ISBN: 978-3-87155-574-9.
- [149] W. Tillmann, P. Hollingsworth, M. Dildorp, A. Eilers u. a.: „Influence of the handling parameters on the characteristics of HVOF sprayed coatings processed by means of fine WC-12Co (2 - 10 µm) powders“. In: *Thermal Spray 2014: Proceedings of the International Thermal Spray Conference*. International Thermal Spray Conference & Exhibition. (Barcelona, Spanien, 21.–23. Mai 2014). Hrsg. von DVS - German Welding Society. Düsseldorf: DVS Media, 2014, S. 324–333. ISBN: 978-3-87155-574-9.
- [150] W. Tillmann und B. Krebs: „Influence of Handling Parameters on Coating Characteristics in Order to Produce Near-Net-Shape Wear Resistant Coatings“. In: *Journal of Thermal Spray Technology* 21 (3-4 2012), S. 644–650. ISSN: 1059-9630. DOI: 10.1007/s11666-012-9735-4.
- [151] W. Tillmann und J. Nebel, Hrsg.: *SFB 708 - 4. öffentliches Kolloquium*. Bd. 4. 3D-Surface Engineering für Werkzeugsysteme der Blechformteilefertigung. Dortmund: Praxiswissen, 2011. 210 S. ISBN: 978-3869750521.

- [152] W. Tillmann und J. Nebel, Hrsg.: *SFB 708 - 5. öffentliches Kolloquium*. Bd. 5. 3D-Surface Engineering für Werkzeugsysteme der Blechformteilefertigung. Dortmund: Praxiswissen, 2012. 194 S. ISBN: 978-3869750712.
- [153] W. Tillmann, E. Vogli, M. Abdulgader, M. Gurrus u. a.: „Particle Behavior During the Arc Spraying Process with Cored Wires“. In: *Journal of Thermal Spray Technology* 17 (5-6 2008), S. 966–973. ISSN: 1059-9630. DOI: 10.1007/s11666-008-9272-3.
- [154] W. Tillmann, E. Vogli, I. Baumann, G. Kopp und C. Weihs: „Desirability-Based Multi-Criteria Optimization of HVOF Spray Experiments to Manufacture Fine Structured Wear-Resistant 75Cr3C2-25(NiCr20) Coatings“. In: *Journal of Thermal Spray Technology* 19 (1-2 2010), S. 392–408. ISSN: 1059-9630. DOI: 10.1007/s11666-009-9383-5.
- [155] W. Tillmann, E. Vogli, I. Baumann und B. Krebs: „Near-Net-Shape and Dense Wear Resistant Thermally Sprayed Coatings“. In: *Key Engineering Materials* 384 (2008), S. 117–123. DOI: 10.4028/www.scientific.net/KEM.384.117.
- [156] W. Tillmann, E. Vogli und B. Krebs: „Influence of the spray angle on characteristics for atmospheric plasma sprayed hard material based coatings“. In: *Journal of Thermal Spray Technology* 17 (5-6 2008), S. 948–955. ISSN: 1059-9630.
- [157] F.-I. Trifa, G. Montavon und C. Coddet: „Integrating a deposition model for off-line spray tools programming“. In: *Thermal Spray 2005: Proceedings of the International Thermal Spray Conference and Exposition*. (Basel, Schweiz, 2.–4. Mai 2005). Düsseldorf: DVS Media, 2005, S. 280–285.
- [158] F.-I. Trifa, G. Montavon und C. Coddet: „Model-Based Expert System for Design and Simulation of APS Coatings“. In: *Journal of Thermal Spray Technology* 16 (1 2007), S. 128–139. ISSN: 1059-9630.
- [159] F.-I. Trifa, G. Montavon und C. Coddet: „On the relationships between the geometric processing parameters of APS and the Al₂O₃TiO₂ deposit shapes“. In: *Surface and Coatings Technology* 195 (1 2005), S. 54–69. ISSN: 0257-8972. DOI: 10.1016/j.surfcoat.2004.07.116.
- [160] F.-I. Trifa, G. Montavon, C. Coddet, P. Nardin und M. Abrudeanu: „Geometrical features of plasma-sprayed deposits and their characterization methods“. In: *Materials Characterization* 54 (2 2005), S. 157–175. ISSN: 1044-5803. DOI: 10.1016/j.matchar.2004.11.010.
- [161] R. Uehara und Y. Uno: „Efficient Algorithms for the Longest Path Problem“. In: *Algorithms and Computation*. Hrsg. von R. Fleischer und G. Trippen. Bd. 3341. Lecture Notes in Computer Science. Springer, 2005, S. 1547–1553. ISBN: 978-3-540-24131-7.
- [162] Z. Ugray, L. Lasdon, J. Plummer, F. Glover u. a.: „Scatter Search and Local NLP Solvers: A Multistart Framework for Global Optimization“. In: *INFORMS Journal on Computing* 19 (3 2007), S. 328–340. ISSN: 1091-9856. DOI: 10.1287/ijoc.1060.0175.

- [163] G.-F. Vander Voort: *Metallography - Principles and Practice*. 4th Edition. Materials Park, Ohio: ASM International, 2007. 752 S. ISBN: 978-0-87170-672-0.
- [164] F. Wahl: *Digitale Bildverarbeitung. Grundlagen, Verfahren, Beispiele*. Berlin, Heidelberg: Springer, 1989. 196 S. ISBN: 3-540-13586-3.
- [165] T. Watanabe: „Correlations between electrode phenomena and coating properties in wire arc spraying“. In: *Thin Solid Films* 316 (1-2 1998), S. 169–173. ISSN: 0040-6090. DOI: 10.1016/S0040-6090(98)00409-X.
- [166] A. Weckenmann, T. Estler, G. Peggs und D. McMurtry: „Probing Systems in Dimensional Metrology“. In: *CIRP Annals - Manufacturing Technology* 53 (2 2004), S. 657–684. ISSN: 0007-8506. DOI: 10.1016/S0007-8506(07)60034-1.
- [167] W. Weißbach: *Werkstoffkunde*. Wiesbaden: Vieweg, 2007. 426 S. ISBN: 978-3-8348-0295-8. DOI: 10.1007/978-3-8348-9417-5.
- [168] T. Wiederkehr, B. Klusemann, D. Gies, H. Müller und B. Svendsen: „An image morphing method for 3D reconstruction and FE-analysis of pore networks in thermal spray coatings“. In: *Computational Materials Science* 47 (4 2010), S. 881–889. ISSN: 09270256. DOI: 10.1016/j.commatsci.2009.11.019.
- [169] T. Wiederkehr, B. Klusemann, H. Müller und B. Svendsen: „Fast, curvature-based prediction of rolling forces for porous media based on a series of detailed simulations“. In: *Advances in Engineering Software* 42 (4 2011), S. 142–150. DOI: 10.1016/j.advengsoft.2011.01.001.
- [170] T. Wiederkehr, A. Kout und H. Müller: „Graphical simulation and visualization of spray-coating processes in computer-aided engineering“. In: *Proceedings of the Vision, Modeling, and Visualization Conference 2008*. (Konstanz, 8.–10. Okt. 2008). 2008, S. 13–20.
- [171] T. Wiederkehr und H. Müller: „Acquisition and Optimization of Three-Dimensional Spray Footprint Profiles for Coating Simulations“. In: *Journal of Thermal Spray Technology* 22 (6 2013), S. 1044–1052. ISSN: 1059-9630. DOI: 10.1007/s11666-013-9927-6.
- [172] T. Wiederkehr und H. Müller: „Efficient Large-Scale Coating Microstructure Formation Using Realistic CFD Models“. In: *Journal of Thermal Spray Technology* 24 (3 2014), S. 1–13. ISSN: 1059-9630. DOI: 10.1007/s11666-014-0194-y.
- [173] T. Wiederkehr, H. Müller, D. Hegels, W. Tillmann und L. Hagen: „Fast Coating Deposition Simulation for Path Planning and Iterative Net-Shape Optimization on Complex Workpieces“. In: *Thermal Spray 2015: Proceedings of the International Thermal Spray Conference and Exposition*. International Thermal Spray Conference. (Long Beach, CA, USA, 11.–14. Mai 2015). Hrsg. von A. Agrawal, G. Bolelli, A. Concustell, Y.-C. Lau u. a. Materials Park, Ohio: ASM International, 2015, S. 390–397. ISBN: 978-1-62708-093-4.

- [174] T. Wiederkehr, H. Müller, B. Krebs und M. Abdulgader: „A deposition model for wire arc spraying and Its computationally efficient simulation“. In: *Proceedings of the International Thermal Spray Conference*. ITSC. (Las Vegas, USA, 4.–7. Mai 2009). Hrsg. von B. Marple, M. M. Hyland, Y.-C. Lau, R. L. Li und G. Montavon. The Printing House Inc., 2009, S. 492–498.
- [175] S. Woop, C. Benthin und I. Wald: „Watertight Ray/Triangle Intersection“. In: *Journal of Computer Graphics Techniques (JCGT)* 2 (1 2013), S. 65–82. ISSN: 2331-7418.
- [176] R. S. Wright, G. Sellers und N. Haemel: *OpenGL Superbible. Comprehensive tutorial and reference*. 6th Edition. Boston: Addison-Wesley Professional, 2013. 848 S. ISBN: 978-0321902948.
- [177] J.-z. Xiang, Y. Zhang, W.-j. Fan, P. Wang und Y.-d. HE: „Spray Deposition Behavior and Numerical Simulation of Growth of Tubular Preform in Spray Forming Process“. In: *Journal of Iron and Steel Research, International* 19 (2 2012), S. 28–35. ISSN: 1006706X. DOI: 10.1016/S1006-706X(12)60056-9.
- [178] K. Yang, M. Liu, K. Zhou und S. Deng: „Recent Developments in the Research of Splat Formation Process in Thermal Spraying“. In: *Journal of Materials* 2013 (2 2013), S. 1–14. ISSN: 2314-4866. DOI: 10.1155/2013/260758.
- [179] J.-T. Yao, J.-Q. Ren, H.-B. Huo, G.-J. Yang u. a.: „Deposition Behavior of Semi-Molten Spray Particles During Flame Spraying of Porous Metal Alloy“. In: *Journal of Thermal Spray Technology* 23 (6 2014), S. 991–999. ISSN: 1059-9630. DOI: 10.1007/s11666-014-0119-9.
- [180] S. Yin, X. Suo, J. Su, Z. Guo u. a.: „Effects of Substrate Hardness and Spray Angle on the Deposition Behavior of Cold-Sprayed Ti Particles“. In: *Journal of Thermal Spray Technology* 23 (1-2 2014), S. 76–83. ISSN: 1059-9630. DOI: 10.1007/s11666-013-0039-0.
- [181] H. Zhang, S. Hu und G. Wang: „Simulation of powder transport in plasma jet via hybrid Lattice Boltzmann method and probabilistic algorithm“. In: *Surface and Coatings Technology* 201 (3-4 2006), S. 886–894. ISSN: 0257-8972.
- [182] T. Y. Zhang und C. Y. Suen: „A fast parallel algorithm for thinning digital patterns“. In: *Communications of the ACM* 27 (3 1984), S. 236–239. DOI: 10.1145/357994.358023.
- [183] J. Zhu, H. Xie, Z. Hu, P. Chen und Q. Zhang: „Residual stress in thermal spray coatings measured by curvature based on 3D digital image correlation technique“. In: *Surface and Coatings Technology* 206 (6 2011), S. 1396–140. ISSN: 0257-8972. DOI: 10.1016/j.surfcoat.2011.08.062.
- [184] Y. Zhu: „Characterization via image analysis of cross-over trajectories and inhomogeneity in twin wire arc spraying“. In: *Surface and Coatings Technology* 162 (2-3 2003), S. 301–308. ISSN: 0257-8972. DOI: 10.1016/S0257-8972(02)00706-5.

- [185] R. Zielke, W. Tillmann, H.-G. Rademacher, M. Abdulgader u. a.: „3D thickness measurement of layer buildup during twin wire arc-spraying process“. In: *Thermal Spray 2014: Proceedings of the International Thermal Spray Conference*. International Thermal Spray Conference & Exhibition. (Barcelona, Spanien, 21.–23. Mai 2014). Hrsg. von DVS - German Welding Society. Düsseldorf: DVS Media, 2014, S. 641–647. ISBN: 978-3-87155-574-9.