THRESHOLDS AND ALGORITHMS IN BAYESIAN INFERENCE

Dissertation

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

der Technischen Universität Dortmund
an der Fakultät für Informatik

von

Lena Krieg

Dortmund

2025

Dekan:
Prof. Dr. Jens Teubner

Gutachter:
Prof. Dr. Amin Coja-Oghlan
TU Dortmund

Prof. Dr. Artur Czumaj
University of Warwick

Datum der mündlichen Prüfung: 29.09.2025

Contents

Acknowledgment

Over the past few years, I have received tremendous support from many people, for which I am very grateful. Without them, this thesis would not have been possible.

First and foremost I would like to thank my supervisor, Amin Coja-Oghlan, for his guidance throughout my studies. Thanks to him I was able to connect with a great research community and attend multiple interesting conferences, summerschools and workshops. I even had the opportunity to co-organize some of them. My last couple of years where filled with interesting discussion, which went beyond only research related topics, including politics, economics, history and many more. He always had an open door to patiently answer any questions we had and was always interested in sharing his mathematical knowledge with us.

I am very grateful to my colleagues for making this journey worthwhile. I would like to thank Olga Scheftelowitsch, my office mate, colleague and friend for her open ear, for our many discussions, for her emotional support and our many great journeys together. I thank Beate Bollig for always supporting and encouraging me, already since I was an undergraduate. I am also thankful for Konstantinos Zampetakis, without whom I would have missed both delicious food and some of the most interesting stories and for my former colleagues, Max Hahn-Klimorth, Jean Bernoulli Ravelomanana and Maurice Rolvien, for sharing their experiences, advices and friendliness that helped me a lot in the beginning. I would also like to thank all my co-authors, especially Lukas Hintze, Haodong Zhu, and Olga Scheftelowitsch, with whom I had the pleasure of working on a great project.

I am also very thankful to my entire family, especially my sister Sara Krieg, whom I very much admire, for always being there for me. Thanks also to my parents for making this journey possible and to Jos Kusiek for accompanying and supporting me on this journey.

Chapter 1

Introduction

Inference Problems represent a fundamental group of problems important in the fields of computer science, physics, mathematics, and statistics. Our ultimate aim is to infer conclusions based on (limited) observations of the real world. We are interested in a *ground truth* that cannot be observed directly but only through indirect, possibly noisy observations. Consider a probabilistic process with some specific but random starting point that evolves over time. In some scenarios we can only observe the outcome of a process but are actually interested in where the process started. Prominent example of this are probabilistic propagation processes in graphs, e.g., the patient zero problem [131, 132]. At the beginning only one person is infected with a disease that then spreads through a population. Only after some time we learn about this new infection and observe the infected individuals at that time. However, we are interested in the start of the infection, in the *patient zero*.

In other scenarios, we can only submit a (possibly limited and noisy) number of queries to reconstruct information using implicit observations only. This is the case if, for example, the observations or measurements themselves are a limited resource. An example is the compressed sensing problem [81, 43, 42] of reconstructing a sparse signal using as few samples, or observations, as possible. One further prominent example is noisy decoding, where we receive a distorted message through a noisy channel but want to recover the sent message [109]. Group testing, the problem of diagnosing infected individuals in a large population using a minimal amount of tests [44], is another example that will accompany us through most of this thesis.

Naturally there are different approaches to model these problems, the two most noteworthy being the Bayesian and frequentist statistics approaches [125]. Both of them have a fundamentally different interpretation of the notion of probability. In *Bayesian statistics*, probability expresses the degree of belief in an event from the viewpoint of the observer. More precisely, the observer assigns each event a value between 0 and 1, where 0 is assigned to impossible events and 1 to known facts (i.e., physical laws, expert knowledge). From this perspective, Bayesian probability is an extension of standard logics, where each statement is either *true* (1) or *false* (0).

In contrast to this, *frequentist statistics* interprets the probability of an event as the limit of its relative frequency in infinitely many trials [75, 125]. However, the binary view of reality still upholds, for each trial, the event either happened or not. To emphasize this difference a bit more, suppose we flip a coin but do not observe the result. Now we ask: What is the probability of the coin showing heads? If we consider this question from the frequentist's perspective, the coin is either showing heads or not. Consequently, the probability of it showing heads is either 0 or 1, but as long as we do not observe it, we do not know which of these is the case. However, if we consider this from the Bayesian perspective, the degree of belief of a coin showing heads is 0.5. In this perspective, the objective reality (that is, either heads or numbers) does not matter, but the observer's degree of belief based on the limited observations.

In contrast to Bayesian statistics, the frequentist's notion of probability does not allow to take any kind of expert or *prior* knowledge into account. Since Bayesian statistics is based on the subjective viewpoint of the observer, prior knowledge about the process is considered. In Bayesian inference, we have some sort of uninformed prior belief $\mathbb{P}[\boldsymbol{\sigma}^*]$ regarding the ground truth $\boldsymbol{\sigma}^*$. Additionally, we receive some observations or evidence $\hat{\boldsymbol{\tau}}$ and want to retrieve an updated version of our belief $\mathbb{P}[\boldsymbol{\sigma}^* \mid \hat{\boldsymbol{\tau}}]$ which we call the *posterior* distribution. The most fundamental step in Bayesian statistics is calculating this via

the Bayes theorem:

$$\mathbb{P}[\boldsymbol{A} \mid \boldsymbol{B}] = \frac{\mathbb{P}[\boldsymbol{B} \mid \boldsymbol{A}]\mathbb{P}[\boldsymbol{A}]}{\mathbb{P}[\boldsymbol{B}]}.$$

This opens the door for a notion of inference that allows the detailed implementation of prior knowledge. However, for practical applications the possibility of calculating the posterior distribution theoretically does not suffice. A naive algorithm to, for example, retrieve the most likely setting according to the posterior distribution takes at least an exponential amount of time, which is definitely not tractable in realistic settings with a huge amount of data. Instead, we are interested in *efficient* algorithms for any inference task. Unfortunately, not only are naive approaches not tractable, but solving these problems in general is computationally hard.

1.1 Why Bayesian Inference is Hard

Given limited observations of the reality, e.g., the outcome of a stochastic process, there exists a multitude of different questions we could consider: What was the most likely starting point of the process? What are different statistical measurements? Can we draw a random sample according to the posterior distribution?

Let us compare this again to a basic task from a frequentist's perspective. Here, one common task is to estimate a small number of parameters of a distribution given a large number of samples. They might be presented with a big amount of data, e.g., numerous measurements of the body height of a coherent population. Now his goal is to retrieve what distribution this data follows. They might assume that the data follows a normal distribution and estimate mean and variance using a maximum likelihood approach. Since the number of parameters is very small compared to the number of samples, we call this *low-dimension* estimation [125]. However, many inference problems in Bayesian statistics are *high-dimensional* estimation problems. For example, we want to estimate the starting point of a stochastic process, which usually is a high-dimensional vector. Here the number of observations we obtain is of the same order as the dimension.

If the sample size is a problem, why don't we just increase the sample size? Let us consider the problem of finding the origin of an infection. Increasing the ratio of observations to parameters to estimate in this setting is only possible if we could run the process *often*, i.e., gather data from a huge amount of similar past endemic events and their origin. The results might give us insights about general infection behaviours or high risk points for beginning epidemics. However, we consider settings where we are presented with one specific observation and want to find the origin for exactly this scenario.

Generally speaking, we are interested in different properties of the posterior distribution. However, all of these problems are generally a part of a famous class of hard problems [38]. More precisely, without restrictions the *calculation problems* above can be proven to #P-hard for many interesting problems like finding the closest codeword [16] or sparse linear regression (both exact and approximate) [96]. To elaborate, we will make a short and simplified excursion into the realm of complexity classes. Formally, we differentiate between *decision* and *counting* problems [10]. The first being problems where the desired output consists of *yes* and *no* answers. The second including problems where we want to *calculate* and output a value.

As an example, we will shortly introduce the decision variant as well as the counting variant of the famous Satisfiability (SAT) problem [37]. We are given a logical formula φ on variables $x_1, \ldots, x_n$ that consists of a conjunction of *clauses*, and each clause again consists of a disjunction of *literals* x_i or $\neg x_i$. The formula $\varphi_{\text{ex}} = (x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_2)$ is a simple SAT formula. We say a configuration *satisfies* φ precisely if it satisfies each clause of φ, thus the assignment $x_1 \mapsto 1, x_2 \mapsto 1$ satisfies φ_{ex}. Given a SAT instance φ the question *"is there a satisfying assignment for φ?"* is the classical decision variant called the SAT problem. The question, how *many* satisfying assignments exist for φ, is the calculation problem called #SAT, as SAT is NP-hard [37].

We call an algorithm *efficient* if it runs in polynomial time in regard to the length of the input. Similarly, we call a problem *efficiently* solvable if there exists an algorithm that solves the problem in polynomial time and call this the class of efficiently solvable problems P. Given a SAT formula φ, without further information we do not know any fundamentally more efficient way to test if φ is satisfiable than to test each and every of the exponentially many assignments [37]. Clearly, this leads to an exponential running time and is therefore not efficient. However, if we are given a single assignment, we can efficiently

verify if this assignment is a *witness* for φ being satisfiable by testing if it satisfies φ and conversely if φ is not satisfiable, there exists no such witness. We call the class of such *efficiently verifiable* problems NP. Further, we call a problem NP-hard if it is *at least as hard* as every problem in NP [37]. Similarly, the class of problems that *count* the witnesses of an NP-problem is called #P[121].

Why we are interested in these classes of problems? These classes characterize the difficulty of the problems contained and even if we can efficiently verify NP-problems there is no known way of solving NP-hard problems efficiently. To make matters worse, it is strongly suspected that there is no efficient way at all. Clearly, #P-hard problems are computationally at least as hard as NP-hard problems. The problems we are interested in here are in a classical sense #P-hard [38, 16]. Consider the question of the most likely starting point of any probabilistic process. The worst case analysis tells us that we know no fundamentally better way than to consider every single path that could lead to the observation we made and compute how likely this happened. At this point, we strongly suspect there exist no general efficient algorithm for Bayesian inference.

1.2 Beyond the Classical Notion of Hardness

Even though this classical perspective might look bleak, research does not end here. We will discuss why classical complexity theory might not give us the final answers.

Consider (noisy) decoding [116] where a *sender* wants to send a binary *information* $x \in \{0,1\}^n$ to the *recipient*. However, the *channel* they use for communication is not flawless and distorts the message randomly. If the sender sends exactly the information she wants to transmit, the receiver won't be able to retrieve exactly the sent information. To overcome this distortion, the sender does not send the true information directly but adds redundancy to her message. More precisely, she sends a (longer) codeword $c \in \mathcal{C}$ of a codebook $\mathcal{C} \subset \{0,1\}^m$. The sender could naively just multiply the message or, better, attach *parity bits* at the end. The decoder now receives a message $\hat{c}$ that has gone through this *noisy* channel and wants to retrieve the true sent codeword c. Generally, they want to find the closest codeword $c \in \mathcal{C}$ to $\hat{c}$.

A worst case analysis indeed tells us that this problem is hard in a classical sense [62]. More precisely, to decode the worst case over the codebooks and received messages is hard. However, in practical settings, this is not really the true question that we are interested in. We actually do not have to deal with the worst case of possible codebooks, but can *choose* codes such that this problem gets solvable efficiently. More precisely, we design codes and corresponding efficient algorithms specially tailored to work with a specific codebook. This perspective was first formalised by Shannon in his famous work on noisy communication [116]. Shannon formalised a rigorous framework of communication through a noisy channel with well defined, *stochastic* errors modelled as random processes. Hence, not every possible change is equally taken into account, but weighted according to their probability. In this perspective finding the closest codeword is indeed not the correct question, but rather finding the most likely codeword according to the received message. This is a stark contrast to settings that deal with *every possible error* and thus a worst case analysis. Shannon provides sharp bounds for the necessary redundancy for different channels that directly come with optimal *Shannon* codes, that consists of randomly distributed codewords [62, 116]. These codes include a minimal amount of redundancy, while coding and decoding in using a Shannon code is in no way efficient. He invented the information theoretical understanding of the concept of entropy as a measure of information content in a message sent through a noisy channel and paved the way for today's understanding of information theory. Famous codes that were invented for this purpose include Hamming codes [64] Reed-Solomon-Codes [107] and many more.

This perspective diverts fundamentally from the classical complexity theoretical viewpoint. Classical complexity theory studies the worst case over all possible inputs, where our problems are in fact hard. There are branches in complexity theory like parametrised algorithms or different analysing methods like average case analysis, amortised analysis [118] or smoothed analysis [117, 49] that have yet another approach beyond worst case analysis. However, we turn this scenario around completely, and *choose* parts of the input that are efficiently solvable and design algorithms specially for these instances.

1.3 Thresholds and Phase Transitions

This perspective raises a multitude of new question. Since we choose the parameters and aim to design an appropriate algorithm, we are interested in the inputs that will actually lead to good results. In the aforementioned scenario of error correcting codes, in reality we aim to minimise the amount of added redundancy [116]. However, if we limit the redundancy too much the received message might not carry enough information so that any estimation will most likely fail [116]. Further, we already know that there exist codes (the worst case scenarios of a classical analysis) that are hard to decode [16]. This leads to multiple questions:

- *When* (for which parameters) is a problem *impossible* to solve?
- *When* is a problem *possible* to solve given no constraints on the computational expenses?
- *When* is a problem solvable *efficiently*?

If we talk about problems that are impossible to solve, we refer to the information theoretic notion of impossibility, where the provided observation does not contain sufficient information to solve the task at hand. Note that these questions are slightly ambiguous as is, since being "solvable" can mean different things in different settings, i.e. we can make different demands on the solution.

In physics, phase transitions refer to the rapid transition between phases that behave fundamentally different e.g. thermodynamic states or magnetisation of iron [94]. Consider a block of iron that does not exhibit magnetic tendencies in room temperature without any prior outer influence. However, if we manipulate it, for example by either cooling it down a lot, it will suddenly show completely different properties and become magnetic [94]. Similarly, aggregate states of matter, say water, changes abruptly on a certain point in temperature and pressure, i.e. water freezes or melts at 0°C. This notion of phase transition carries over to probabilistic structures [94]. Generally, we have a large probabilistic model that comes with parameters (e.g. the noise level, the size and form of the observation) with changing behaviour according to the choice of the parameter. Usually, the parameter models local (microscopic) interaction in the model. However, some macroscopic properties of the outcome change according to the chosen parameter. For some properties there exists a *transition* at one specific point of the parameters, where the property changes dramatically. More precisely, within a phase the macroscopic behaviour changes such that we can interpolate the behaviour. However, on a phase transition we cannot interpolate the behaviour beyond this point. Consider water in its liquid state, we can interpolate how it behaves at in the temperature range where it stays liquid based on a few observations in this range. However, ones it freezes, this interpolation fails and the behaviour changes completely. This behaviour is characterised by a measure, like the free energy of the system, being analytical within a phase but non-analytic at these points of rapid change called *phase transitions* [94].

The observed properties can describe different phenomenons, like the geometry of a solution space. However, the property we are interested in is the answer to the questions above: Whether it is possible to solve a given (inference task) and if this is efficiently possible.

Heuristic arguments and experiments indicate that most, if not all, inference problems go through a transition from easy over hard to impossible [129] as their parameters changes. We are mostly interested in the following three phases:

- an easy phase, were inference is efficiently possible,
- a hard phase, where inference is theoretically possible but this problem is computationally hard,
- and the impossible phase, where inferring is impossible, i.e. every algorithm fails with high probability (w.h.p., with probability approaching one as the size of the input tends to infinity).

In different problems these *phase transitions* have been extensively studied [101, 129, 1, 4, 51, 55, 40]. Naturally, researchers are interested in the characterisation of the hard phase, if existent. We will discuss this in more detail in Section 1.4.

If we direct our attention to the research field of statistical physics, there are a lot of approaches for solving this type of problems by combining probabilistic constructions with message passing algorithms like *Belief Propagation* on random factor graphs [129]. These approaches heavily rely on physics intuition but are only partially rigorously proven. However, many of physicists' "predictions" can be actually

studied and confirmed rigorously [41, 95]. Sometimes these physicists' techniques can be used to find even *efficient* algorithms that are optimal, i.e. solve the problems for the whole (possible) regime of parameters.

An example of an inference problem where one can observe these phase transitions as predicted by physicists' intuition is noiseless decoding [116] that coincides with the planted random k-XORSAT problem. A k-XORSAT formula is similar to a k-SAT formula, with the difference that each clause consists of XOR-operators $\oplus$ instead of the disjunctions in k-SAT [101]. Here, a clause $c = \ell_1, \dots, \ell_k$ is satisfied if the number of "true" literals is odd. Given a k-XORSAT formula it is easy to tell if it is satisfiable and to give a solution (if existent) by Gaussian elimination [94]. A *random* k-XORSAT formula on n variables and $m = dn/k$ clauses with the density d is drawn by choosing independently random XORSAT clauses of length k. We *plant* a random variable assignment $\boldsymbol{\sigma}^*$ by sampling the k-XORSAT formula φ conditioned on being satisfied by $\boldsymbol{\sigma}^*$.

This scenario fits seamlessly in our notion of inference problems, where we consider the task to infer the planted solution $\boldsymbol{\sigma}^*$ given φ [116]. Given φ, each satisfying assignment is equally likely to be the planted solution. Figure 1.1 visualizes the solution space for a planted k-XORSAT formula. On the left, when there are only few clauses, the solutions build a giant cluster, i.e. there exists many similar solutions. We expect finding an arbitrary solution to be easy, since there are many solutions and the solutions are clustered, i.e. from the outside there is a "clear" direction of optimising a "non-solution" [56, 57]. In this phase, we expect local algorithms to work that start at a random configuration and improve locally to find a solution.

If we add more clauses, the cluster shatters into many *small* clusters of solutions that are far apart from another [11]. In this phase we do not expect local algorithms to find an arbitrary solution to work properly [55]. If a local algorithm tries to optimise a configuration and might get stuck in a local optimum without a solution. As we increase the number of clauses, the solutions space shrinks further. Once we exceed the satisfiability threshold α_{sat} for random k-XORSAT, we are left with one cluster in phase three. In this regime a completely random k-XORSAT formula would have no solution w.h.p. [11, 101], but since we conditioned the formula on being satisfiable, this cluster remains. However, through the planted solution, this cluster shrinks down to precisely the planted solution.

If we consider the question, whether the planted solution is inferable, we observe different behaviour corresponding to the visualised phases. For the low density phase on the left of Fig. 1.2 we cannot hope to gather any information about the planted solution, since the number of solutions is too large [11]. More precisely, in phase one and two of Fig. 1.2, if we sample a random solution to φ we are unlikely to sample any solution nearby $\boldsymbol{\sigma}^*$. Hence, the variables where a sampled solution and $\boldsymbol{\sigma}^*$ agree on is small, they have a *small overlap*.

If we sample a random solution in the third phase, we will find a solution nearby $\boldsymbol{\sigma}^*$, that diverts from $\boldsymbol{\sigma}^*$ only for a small number of variables [11]. In this phase *approximate* recovery, finding a solution with *high overlap* to $\boldsymbol{\sigma}^*$ is possible. In the last phase, the $\boldsymbol{\sigma}^*$ surrounding cluster shrinks, such that the probability mass gets centralised solely on $\boldsymbol{\sigma}^*$. This happens precisely when the 2-core, the maximal subgraph where each vertex has degree at least two, includes the whole graph. This coincides precisely with the point where every vertex has degree at least two w.h.p. at $\alpha_{\text{ex}} = \Theta(\ln(n))$ [52]. Here recovering $\boldsymbol{\sigma}^*$ exactly without any mistake w.h.p. is possible.

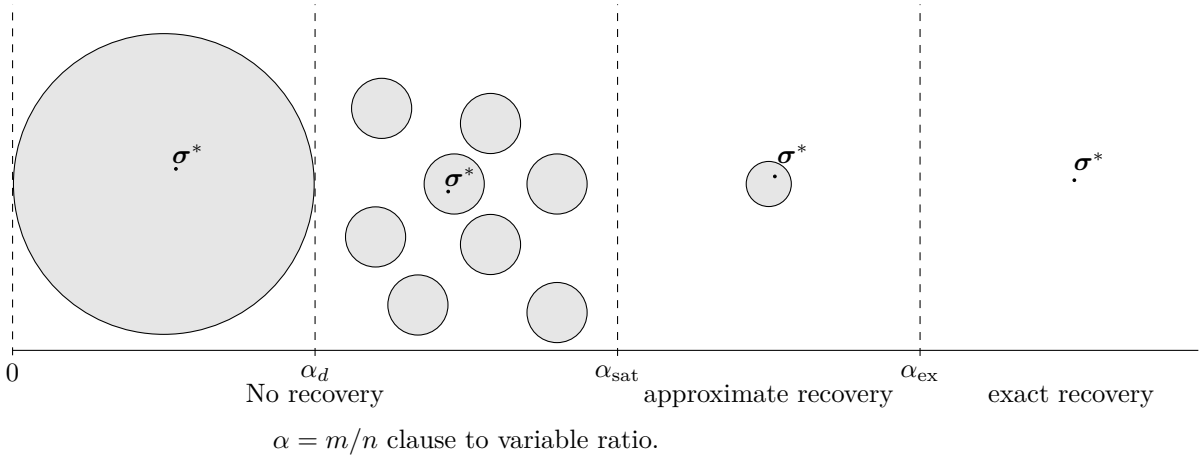


Figure 1.1: Pictogram of solutions space of random planted k-XORSAT.

If we divert from planted random k-XORSAT to the *null model* , a completely random k-XORSAT formula without a planted solution, we retrieve a slightly different geometry visualized in Fig. 1.2. The density α_d marks the point where the connected component shatters into smaller, not connected components with a large distance. Further, since we haven't conditioned φ on being satisfiable, once we surpass a threshold α_{sat} the formula φ becomes unsatisfiable w.h.p. Quite interestingly, the state space of the first two phases of Fig. 1.1 and Fig. 1.2 look alike and indeed, the total variation distance between these two distributions is $o(1)$ [11]. Hence, given k-XORSAT formula φ for $\alpha < \alpha_{\text{sat}}$, it is impossible to tell if φ was drawn as a planted k-XORSAT formula or a completely random k-XORSAT formula. Only when we exceed the satisfiability threshold we can distinguish between the planted and the null model [11]. Note that strictly speaking random k-XORSAT is no optimal example, since the existence of Gauss elimination as an efficient algorithm contradicts the existence of a proper hard phase – it merely serves as an illustration of the coinciding transitions of the geometry of the state space.

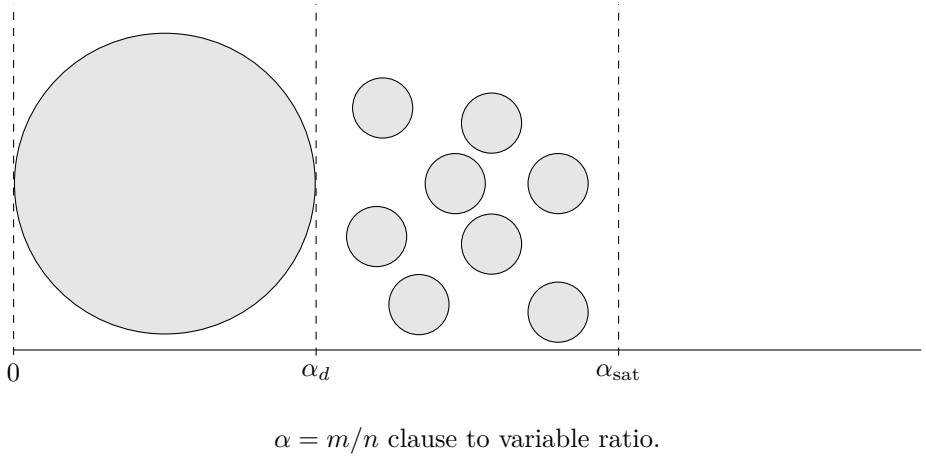


Figure 1.2: Pictogram of solutions space of random k-XORSAT as found in [94].

Another prime example of a planted structure where we actually observe the whole range of phases is the *stochastic block model* ('SBM') [66]. It was first introduced in the field of social network analysis [67] and serves as a benchmark generative model of community detection in networks [1, 91]. While there are different variants of the SBM in literature, we are interested here in the *disassortative* SBM [39]. In this model, we consider n vertices $v_1, \dots, v_n$ randomly partitioned in q groups. Then we build a random graph such that two vertices in the same cluster are less likely to be connected than two vertices in different groups parametrised by the *inverse temperature* parameter β and a parameter $d = d(\beta)$ for

the average degree. We follow the definition of [36] and define

$$d_{\text{in}} = \frac{dq\exp(-\beta)}{q-1+\exp(-\beta)}, \qquad d_{\text{out}} = \frac{dq}{q-1+\exp(-\beta)}. \tag{1.3.1}$$

Given the assignment $\boldsymbol{\sigma}^*$ two vertices v, w of the emerging graph $\boldsymbol{G}_{\text{SBM}}$ have the edge probability

$$\mathbb{P}[\{v,w\} \in E(\boldsymbol{G}_{\text{SBM}}) \,|\, \boldsymbol{\sigma}^*] = \begin{cases} \min(1, d_{\text{in}}/n) & \text{if } \boldsymbol{\sigma}^*_v = \boldsymbol{\sigma}^*_w \\ \min(1, d_{\text{out}}/n) & \text{else} \end{cases} \tag{1.3.2}$$

Intuitively, a higher temperature (lower β) leads to a more "entropic" system whereas a lower temperature (higher β) is associated with a more ordered system. In the SBM, a higher temperature leads to more monochromatic edges (edges between vertices of the same partition), and similarly a lower temperature leads to fewer, up to no monochromatic edges [2]. If the temperature would approach $\beta \to \infty$ we would expect to see three clusters with no connection at all inside each cluster, whereas for $\beta \to 0$ we expect the result to resemble an Erdős–Rényi-graph [1].

Again, we consider the assignments of the variables to the corresponding cluster as the planted solution $\boldsymbol{\sigma}^*$. The goal is to infer an non-trivial estimate of $\boldsymbol{\sigma}^*$ from the emerging graph $\boldsymbol{G}_{\text{SBM}}$. If we fix the temperature $\beta > 0$, this task gets easier as d increases, since more edges are observed and thus more information, and harder for smaller d.

For d surpassing the *Kesten-Stigum threshold* [80]

$$d_{\text{KS}}(\beta) = \left(\frac{q-1+\exp(-\beta)}{1-\exp(-\beta)}\right)^2. \tag{1.3.3}$$

Abbe and Sandon [3] contributed an algorithm that recovers a high overlap solution to $\boldsymbol{\sigma}^*$. For $q = 2$ this thresholds coincides with the information theoretic threshold [92], i.e. the threshold $d_{\text{IT}}(\beta)$ below which it is impossible to recover a non-trivial solution. However, for $q \geq 5$ there exists a gap between those thresholds [36]. There is no known efficient algorithm that is able to recover $\boldsymbol{\sigma}^*$ in this middle phase and hence this phase $d_{\text{IT}}(\beta) < d < d_{\text{KS}}(\beta)$ was conjectured to be hard, i.e. that efficient recovering $\boldsymbol{\sigma}^*$ in this phase is not possible at all [39]. Further, the Kesten-Stigum threshold was one of the first sharp thresholds proven to be characterized by low-degree polynomial algorithms [69] as described in Section 1.4.

However, even though in these regimes it is efficiently possible to recover a high overlap solution, a different question is, if the MAP strategy for recovering $\boldsymbol{\sigma}^*$ is viable. In a recent work, we proved for a variant of the assortative SBM that even though the probability mass is centralized around $\boldsymbol{\sigma}^*$ there exists solutions that have nothing to do with $\boldsymbol{\sigma}^*$ and surpass $\boldsymbol{\sigma}^*$ in terms of a posteriori likelihood [35]. This phenomenon is visualised in Fig. 1.3, where we clearly see the probability mass centralised around $\boldsymbol{\sigma}^*$, however there exists some configurations with higher a posteriori likelihood than $\boldsymbol{\sigma}^*$, even though these configuration do not carry much probability mass in total.

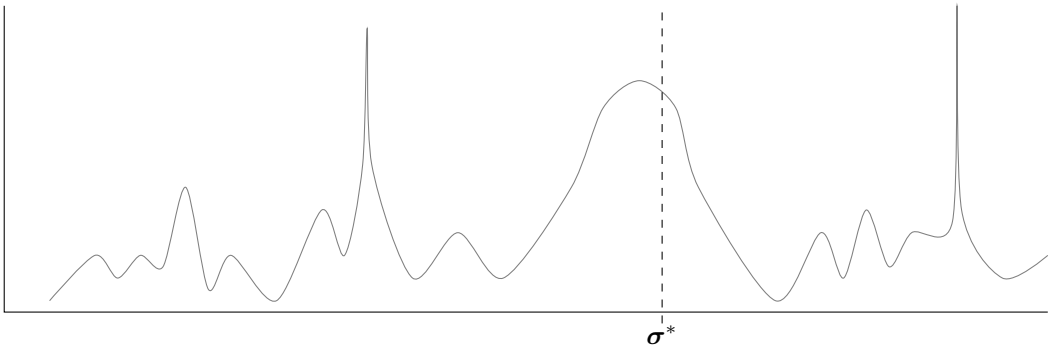


Figure 1.3: Schematic visualisation of probability landscape in the SBM

This physicists' perspective is going to help us studying phase transitions in multiple different settings, including in group testing. Group testing as introduced by [44] does not only pose as a prime example of

inference problems, but also as a representative of a group of problems that are easy to state but hard to solve. Consider a large population where some individuals are infected with a disease. We aim to identify exactly who is infected by using pooled tests, where possibly multiple individuals participate per test. A test is supposed to be positive if at least one of the participating individuals is infected, however there is a chance that a tests returns a wrong result. Moreover, we want to minimise the number of tests used, as each test might be expensive in realistic settings. There exists a threshold of tests needed to be able to diagnose the population correctly with high probability (w.h.p., i.e. with probability approaching 1 with increasing number of individuals). Of course, this *threshold* depends on the specific model and models' parameters, e.g. the number of infected individuals or the amount of noise. This work does not only study group testing in different setting, but we are also going to use it as a prime example for some fundamentals in Chapter 2.

1.4 Low Degree Algorithms

As already seen above, the easy-hard threshold has been studied in many different problems. Naturally, researchers aim to understand the fundamental structural difference between easy and hard problems. Consider some optimisation problems in random instances and their gap between optimum and algorithmically reachable solution. The simplest random graph model, a sparse random Erdős–Rényi graph $G(n, d/n)$, will contain an independent set of sizes up to $2(\ln(d)/d)$ w.h.p. [53]. However, the largest independent set that is known to be reachable algorithmically has size $\ln(d)/d$, and structural evidence suggests that there is no algorithm achieving a better result [4, 29]. The algorithmically reachable value coincides exactly with the value achievable by local algorithms [56, 106]. There are similar examples for approximate message passing, an iterative message passing approach based on local interactions, that is known to not achieve optimal results in tensor principal component analysis [108].´ These different algorithmic approaches separately do not capture the whole picture and thus are not sufficient to characterize this threshold. Hence, on the search for the best evidence of a problem being hard lead to a unified class of algorithms that include the one above. These class of *low degree algorithms* restricts us to algorithms that only calculate low degree polynomial functions. More precisely, when we get a high dimensional input we are restricted to polynomial function of degree at most $O(\ln(n))$ with n being the dimension.

For a wide range of problems these low degree algorithms are actually known to be precisely as powerful as the best known poly-time algorithms, hence it was conjectured that low degree algorithms are *precisely* as powerful as all poly-time algorithms for natural high-dimensional problems.

1.5 Summary of Results

The present thesis contains results for group testing , random k-XORSAT and the patient zero problem.

Group Testing as proposed by Dorfman [44] is the problem of finding k infected individuals among a large set of n individuals using as few group tests as possible. In Section 3.1 the most common settings of the group testing problem are listed. The contained results solve the group testing problem for multiple settings completely, i.e. give an exact threshold of tests needed for identifying the set of infected individuals w.h.p. as well as providing efficient algorithms that achieve this. The result taken from [65], included as Appendix C, covers the complete analysis for the linear noisy group testing in Section 3.5 including a precise bound for the number of necessary tests for both adaptive and non adaptive group testing as well as testing schemes and efficient algorithms matching these bounds. Moreover, the results taken from [33] included as Appendix B provide the precise threshold for approximate recovery for sublinear noisy group testing. This includes a test design with the optimal number of tests as well as a design for exact recovery complemented by *efficient* decoding algorithms and is covered in more detail in Section 3.4. For exact recovery this work includes a lower bound of the number of needed tests for a class of test designs that achieved the best results so far.

In Random k-XORSAT one draws a random k-XORSAT formula $\Phi_{d,k}$ containing dn clauses consisting of exactly k random literals. Below the k-XORSAT satisfiability threshold d_k a random k-XORSAT

formula $\Phi_{d,k}$ is satisfiable w.h.p. and above it is unsatisfied w.h.p. The 3-XORSAT threshold was first proven by Dubois and Mandler [46] and more than ten years later generalized to k-XORSAT using a very complicated and computer assisted proof by Pittel and Sorkin [101]. The first generalized proof that does not rely on computer assistance was later published [11]. The results from [34], included here as Appendix D and more detailed discussed in Chapter 4 include a re-proof of this threshold [11]. The stated new proof relies on a fundamental combinatorial understanding of the random k-XORSAT problem and more precisely the influence of local interactions for the global behaviour. Additionally, the re-proof contains a generalization of the original threshold to random matrices over $\mathbb{F}_q$.

The Patient Zero Problem generally describes the task of finding the source of an infection process in a graph based on an observation made after the infection already spread in the graph for some time [131]. This problem comes in a multitude of settings. The infection process itself depends on multiple parameters, like infection rate, recovery rate, time, or the information contained in the observation [19, 77, 79, 131]. Also, different underlying graphs can be studied for infection processes, like different random graphs. The results here are taken from [14], included as Appendix E. For a rather simple model, the independent cascade model [78, 130] on random trees, the present result give a sharp threshold for the infection parameters for recovering the spreading' source. Additionally, the work gives a simple but efficient estimator that returns the most likely infection source.

Chapter 2

Basics

Before we move forward with discussing the proof ideas and results, we formalize the generalized setting of a Bayesian inference problem using the teacher student model. Further, we introduce the basic techniques to model inference problems using factor graphs and the general Belief Propagation algorithm. Through the entirety of this chapter, we are going to use the group testing problem as an ongoing example.

2.1 Group Testing - a short Introduction

Group testing, first introduced formally by Dorfman [44] is the problem of finding k infected individuals among a large set of n individuals using as few pooled tests as possible. There exists a multitude of settings for group testing, for which the most important ones are introduced later in Section 3.1. Here we introduce the setting of noisy probabilistic non adaptive group testing in the linear regime as an example [8]. Consider a population of n individuals, each individual is independently infected with a constant probability $\alpha \in (0,1)$. These states are given in an infection vector $\boldsymbol{\sigma}^* \in \{0,1\}^n$ where $\boldsymbol{\sigma}_i^* = 1$ indicates that individual i is infected. We conduct m pooled tests in parallel, where each test contains a set of individuals and is *actually* positive if at least one of the contained individuals is infected and negative otherwise. Since the tests are conducted in parallel the test design is fixed before testing. We model a *test design* as a bipartite factor graph $G = (V, F, E)$ where V represents the individuals and F represent the tests. Each test $a \in F$ is connected to every individual that participates in said test. The actual test results can be represented as a vector $\boldsymbol{\tau}'$ with $\boldsymbol{\tau}'_a = \mathbb{1}\left\{\sum_{i \in \partial a} \boldsymbol{\sigma}_i^* \geq 1\right\}$. However, the assumption of observing this perfect result is not realistic. Instead, it is usually assumed that each test has an independent chance of showing a false positive or negative result [63, 72, 87]. Here, each actually positive test is observed positively with probability p_{11} and shows a false negative result with probability $p_{10} = 1 - p_{11}$. Similarly, an actually negative test gives a correct negative result with probability p_{00} and a false positive one with probability $p_{01} = 1 - p_{00}$.

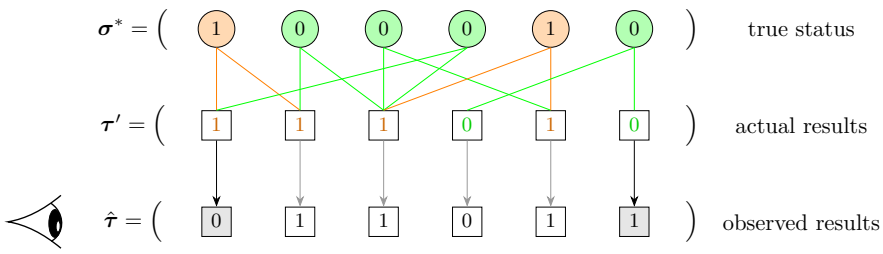


Figure 2.1: Visualization of group testing. The round vertices visualize the individuals and the rectangle vertices the grouped tests. If an individuals is connected to a test, it participates in the grouped test. The grey filled tests mark observed tests which incorrect test result that divert from the actual results.

2.2 Teacher Student Model

The teacher student model is a general framework that puts inference problems in a formalised setting. It was first introduced in 1989 [58], we follow [129]. Assume a teacher generates a ground truth $\boldsymbol{\sigma}^*$ following to a distribution $\mathbb{P}[\boldsymbol{\sigma}^*]$. Now the teacher runs a probabilistic process with $\boldsymbol{\sigma}^*$ as a starting point and draws an observation $\hat{\boldsymbol{\tau}}$ distributed as $\mathbb{P}[\hat{\boldsymbol{\tau}} \mid \boldsymbol{\sigma}^*]$ modelling this process. Then the teacher gives $\hat{\boldsymbol{\tau}}$ as well as some information about $\mathbb{P}[\boldsymbol{\sigma}^*]$ and $\mathbb{P}[\hat{\boldsymbol{\tau}} \mid \boldsymbol{\sigma}^*]$ to the student. Finally, the student aims to infer the ground truth $\boldsymbol{\sigma}^*$ based on this information. Naturally their best strategy is to calculate the posterior $\mathbb{P}[\boldsymbol{\sigma}^* \mid \hat{\boldsymbol{\tau}}]$. Knowing Bayes theorem the student calculates

$$\mathbb{P}[\boldsymbol{\sigma}^* \mid \hat{\boldsymbol{\tau}}] = \frac{\mathbb{P}[\boldsymbol{\sigma}^*]\mathbb{P}[\hat{\boldsymbol{\tau}} \mid \boldsymbol{\sigma}^*]}{\mathbb{P}[\hat{\boldsymbol{\tau}}]} \propto \mathbb{P}[\boldsymbol{\sigma}^*]\mathbb{P}[\hat{\boldsymbol{\tau}} \mid \boldsymbol{\sigma}^*].$$

If the student now wants to return the best possible guess, naturally they search for the configuration that maximises the above measure. Remember that $\hat{\boldsymbol{\tau}}$ originated from a process the teacher conducted. However, we could in principle try to draw random "observation" $\hat{\boldsymbol{\tau}}_{\text{null}}$ that is not based on any real truth. This is called a *null model*, and for itself does not carry any type of data about any ground truth. However, the distribution that the teacher draws the observation from often is just be the null model weighted by the ground truth according to a weight function $\psi_{\hat{\boldsymbol{\tau}}}$, i.e.

$$\mathbb{P}[\hat{\boldsymbol{\tau}} = \hat{\tau} \mid \boldsymbol{\sigma}^*] \propto \mathbb{P}[\hat{\boldsymbol{\tau}}_{\text{null}} = \hat{\tau}]\psi_{\hat{\tau}}(\boldsymbol{\sigma}^*) \tag{2.2.1}$$

where $\propto$ means equal up to normalisation. If the student combines (2.2.1) with Bayes theorem, they obtain

$$\mathbb{P}[\boldsymbol{\sigma}^* \mid \hat{\boldsymbol{\tau}}] \propto \mathbb{P}[\boldsymbol{\sigma}^*]\psi_{\hat{\tau}}(\boldsymbol{\sigma}^*). \tag{2.2.2}$$

This equation is known as the *Nishimori* property or a Bayes optimality [36, 129] that allows the student to focus on $\psi(\boldsymbol{\sigma}^*)$.

Let us now put the group testing process as above in the setting of the teacher student model. The teacher draws the ground truth, the true infection vector $\boldsymbol{\sigma}^*$ follows $\mathbb{P}[\boldsymbol{\sigma}^*] = \alpha^{\#\text{infected}}(1-\alpha)^{\#\text{not infected}}$ for a known constant α. Then, the teacher takes the test design G and performs the tests accordingly. Finally, the teacher tells the student the infection probability α and hands over the test design G together with the test results $\hat{\boldsymbol{\tau}}$. Given the test design and $\hat{\boldsymbol{\tau}}$ the student tries to infer the ground truth $\boldsymbol{\sigma}^*$ based on these information. Of course, the hardness/possibility of the students task heavily depends on the parameters $\boldsymbol{p}, \alpha$ and the test design G.

Further, in group testing, our null model consists of random test results. To be a bit more precise, given an infection parameter α, each test independently "draws" their edges to be connected to an infected individual with probability α. Then, the test declares itself as actually positive iff at least one of the drawn edges is "infected" flips according to the noise. Note, that the tests do not maintain any kind of consistency towards the other tests, hence this could lead to an impossible test result in the noiseless case. However, here the (2.2.1) is indeed satisfied, as the teacher just weights the random test result according to probability based on $\boldsymbol{\sigma}^*$. In noiseless group testing, the teachers weight function plainly excludes every observation that disagrees with the "correct" test results. In the noisy setting the null model is weighted according to the total probability that the correct test result "flipped" resulting in $\hat{\tau}$.

Factorizing Measures We are usually interested in high dimensional variables $\boldsymbol{\sigma}^* \in \Omega^n$ and $\hat{\boldsymbol{\tau}} \in \hat{\Omega}^m$ for some state spaces $\Omega, \hat{\Omega}$ and dimensions n, m. Furthermore we are particularly interested in measures where $\hat{\boldsymbol{\tau}}_i$ does not depend on the complete ground truth, but only a small part of it. Let therefore ∂j be the set of indices that $\hat{\boldsymbol{\tau}}_j$ depend on. We say a measure factorize, if it can be written as follows;

$$\mathbb{P}[\boldsymbol{\sigma}^* = \sigma] = \prod_{i=1}^{n} \mathbb{P}[\boldsymbol{\sigma}_i^* = \sigma_i]$$

$$\mathbb{P}[\hat{\boldsymbol{\tau}} = \hat{\tau} \mid \boldsymbol{\sigma}^* = \sigma] = \prod_{i=1}^{m} \mathbb{P}[\hat{\boldsymbol{\tau}}_j = \hat{\tau}_j \mid \boldsymbol{\sigma}^* = \sigma] = \prod_{i=1}^{m} \mathbb{P}[\hat{\boldsymbol{\tau}}_j = \hat{\tau}_j \mid \boldsymbol{\sigma}^*_{\partial i} = \sigma_{\partial i}]$$

The student now takes $\hat{\boldsymbol{\tau}}$ as well as whatever information about the distributions $\mathbb{P}[\boldsymbol{\sigma}^*]$ and $\mathbb{P}[\hat{\boldsymbol{\tau}} \mid \boldsymbol{\sigma}^*]$ that they receive from the teacher and tries to infer $\boldsymbol{\sigma}^*$. Naturally, the best they can do is to use Bayes' theorem and calculate $\mathbb{P}[\boldsymbol{\sigma}^* \mid \hat{\boldsymbol{\tau}}]$. Hence, the student calculates

$$\begin{aligned}\mathbb{P}[\boldsymbol{\sigma}^* = \sigma \mid \hat{\boldsymbol{\tau}} = \hat{\tau}] =& \frac{\mathbb{P}[\hat{\boldsymbol{\tau}} = \hat{\tau} \mid \boldsymbol{\sigma}^* = \sigma]\mathbb{P}[\boldsymbol{\sigma}^* = \sigma]}{\mathbb{P}[\hat{\boldsymbol{\tau}} = \hat{\tau}]} \\ &\propto \prod_{i=1}^{n} \mathbb{P}[\boldsymbol{\sigma}_i^* = \sigma_i] \prod_{j=1}^{m} \mathbb{P}\big[\hat{\boldsymbol{\tau}}_j = \hat{\tau}_j \,\big|\, \boldsymbol{\sigma}^*_{\partial j} = \sigma_{\partial j}\big]. \end{aligned} \tag{2.2.3}$$

For Group Testing each individual is infected independently of all other individuals with probability α. Moreover, each test result only depends on the contained variables. In this case for every test j the set ∂j indicates the set of included individuals. Hence, taken from [33], we have

$$\begin{aligned}\mathbb{P}[\boldsymbol{\sigma}^* = \sigma] =& \prod_{i=1}^{n} \mathbb{P}[\boldsymbol{\sigma}_i^* = \sigma_i] = \prod_{i=1}^{n} \alpha^{\mathbb{1}\{\sigma_i=1\}}(1-\alpha)^{\mathbb{1}\{\sigma_i=0\}} = \alpha^{\#\text{infected}}(1-\alpha)^{n-\#\text{infected}} \\ \mathbb{P}[\hat{\boldsymbol{\tau}} = \hat{\tau} \mid \boldsymbol{\sigma}^* = \sigma] =& \prod_{i=1}^{m} \mathbb{P}[\hat{\boldsymbol{\tau}}_j = \hat{\tau}_j \mid \boldsymbol{\sigma}^*_{\partial i} = \sigma_{\partial i}] \\ =& \prod_{i=1}^{m} \mathbb{1}\left\{\sum_{i\in\partial j} \sigma_i > 0\right\} p_{1\hat{\tau}_j} + \mathbb{1}\left\{\sum_{i\in\partial j} \sigma_i = 0\right\} p_{0\hat{\tau}_j}. \end{aligned}$$

Mismatched Inference Problems If the student does not get enough and or correct information from the teacher, we have a *mismatched* case in which (2.2.1) is not satisfied [12]. More precisely, the 'true' weight function ψ and the prior $\mathbb{P}[\boldsymbol{\sigma}^*]$ is distorted and varies from the information the student receives. At first glance this might seem like an inappropriate, but in realistic settings this might as well be the case. First, the assumption the student makes or the information they get about the distributions might just be not correct. More precisely, parameters for the distribution might either be incorrect or not available precisely since realistically distributions and parameters are usually *estimated* and hence have a error margin – even if a small one. Research on this is still very much limited including basic studies from statistical physics [12, 22, 122] as well as research on mismatched estimations of rank one matrices [102, 103].

In real world pooled tests for group testing the specificity and sensitivity of a test is also only the result of a large scale study of exactly this test. However, every estimate is bound to possibly have an error margin, and thus the information we are presented with in group testing might not be precise. This also goes for the infection probability α, which realistically is also impossible to give precisely. Section 3.5.4 includes a result from [65] that covers a mismatched prior for group testing.

2.3 Factor Graphs

An important tool to model aforementioned factorized inference problems are *factor graphs* [94, 82]. A factor graph is a bipartite graph $G = (V, F, E)$,where V are called variable nodes and F factor nodes. Given a factorizing measure over a state space Ω^n each variable node represents one dimension of $\boldsymbol{\sigma}^*$. Each factor node $a \in F$ is assigned a *weight function* $\psi_{a,\hat{\tau}} : \Omega^{\partial a} \to \mathbb{R}_{\geq 0}$. This weight function models the interaction of the neighbouring variables ∂a. Intuitively, a lower weight corresponds to a less likely configuration of the factor nodes neighbourhood with $\psi_{a,\hat{\tau}}(\sigma_{\sigma a}) = 0$ being the extremal case of an impossible configuration. Accordingly, a higher weight corresponds to a more likely scenario of the neighbours configuration. With this we define the *Gibbs measure* μ as well as the partition function $Z_{\hat{\tau}}$ following [94]

$$\psi_{\hat{\tau}}(\sigma) = \prod_{a \in F} \psi_{a,\hat{\tau}_a}(\sigma_{\partial a})$$
$$Z_{\hat{\tau}} = \sum_{\sigma \in \Omega} \psi_{\hat{\tau}}(\sigma)$$
$$\mu_{\hat{\tau}}(\sigma) = \frac{\psi_{\hat{\tau}}(\sigma)}{Z}.$$

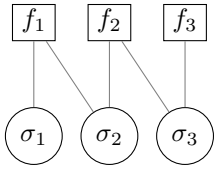


Figure 2.2: Depiction of a factor graph for $\psi(\sigma) = f_1(\sigma_1, \sigma_2) f_2(\sigma_2, \sigma_3) f_3(\sigma_3)$

Given a factorizing measure, a corresponding factor graph is not unambiguous. However, every factor graph "encodes" a basic property of the probability distribution. That is that every two "well-separated" sets of variables interact only through variables between those sets. This property is called *global Markov property* and can be formalized as follows:

Proposition 2.3.1 (global Markov property (Proposition 9.2 of [94]))**.** *Let $G = (V, F, E)$ be a factor graph and $A, B, S \subseteq V$ three sets of variables such that S separates A from B, i.e. every path from a variable in A to another variable in B in G passes through at least one variable in S, then*

$$\mathbb{P}[\boldsymbol{\sigma}_A^*, \boldsymbol{\sigma}_B^* \mid \boldsymbol{\sigma}_S^*] = \mathbb{P}[\boldsymbol{\sigma}_A^* \mid \boldsymbol{\sigma}_S^*] \mathbb{P}[\boldsymbol{\sigma}_B^* \mid \boldsymbol{\sigma}_S^*] \tag{2.3.1}$$

We call A and B conditionally independent on S.

In Group Testing the factor graph is directly given by the test design G, where the individuals correspond to the variable nodes and the test to the factor nodes. A test assigns a configuration a weight corresponding to the flip probability. Consider a positively displayed test a, then either a is correctly displayed positively (and has not flipped) or it rendered a falsely positive test result. The former is the case if there at least one infected individual in a's neighbourhood and then happens with probability p_{11}. Similarly, the latter happened when there is no infected individual in a's neighbourhood and then happens with probability p_{01}. Hence a assigns a configuration the weight p_{11} if it contains an infected individuals in a's neighbourhood and p_{01} else. In the scenario this gives the weight function for each test as follows [8]:

$$\psi_{a,\hat{\tau}_a}(\sigma_{\partial a}) = \mathbb{1}\left\{\sum_{i \in \partial a} \sigma_i > 0\right\} p_{1\hat{\tau}_a} + \mathbb{1}\left\{\sum_{i \in \partial a} \sigma_i = 0\right\} p_{0\hat{\tau}_a}$$

Additionally, for the case of linear probabilistic group testing introduced above, we have a prior α for each individual. Hence, one could add another factor node f_i for each individual i that assigns a configuration weight α if i is infected and $(1 - \alpha)$ else:

$$\psi_{f_i,\hat{\tau}_a}(\sigma_{\partial a}) = (1 - \alpha)^{1-\sigma_i} \alpha^{\sigma_i}.$$

For the special case of noiseless group testing ($p_{11} = p_{00} = 1$), the weights of the factor nodes impose a deterministic constraint on σ excluding configurations that are not compatible with the observed results. More precisely, a factor node of a positive test result would assign a neighbouring configuration without an infected individual the weight 0 and a factor node corresponding to a negative test result would exclude configurations with an infected individual in this test. Similarly, $Z_{\hat{\tau}}$ in this setting is precisely the sum of configurations that are compatible with an observation $\hat{\tau}$ weighted by their prior.

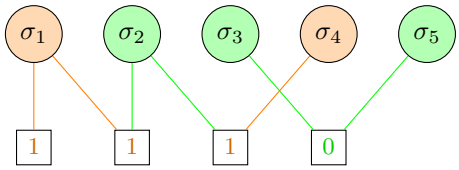


Figure 2.3: Factor graph for a noiseless group testing setting. The true infection state is colour coded in the round vertices, albeit not known to the factor graph or a potential decoding algorithm.

Figure Fig. 2.3 depicts a example factor graph for noiseless group testing. For the sake of simplicity we do not consider the prior of the individuals. Since the weights of the tests act as a constraint to the configuration, for a valid configuration the weight comes to $\psi_{\hat{\tau}}(1,0,0,1,0) = 1$ and for an invalid one to $\psi_{\hat{\tau}}(1,0,0,1,1) = 0$ (note that the last test is violated). The results fix the configurations of $\sigma_1, \sigma_3, \sigma_5$ but only tells us that at least one of σ_2, σ_4 is infected. Hence, unweighted by the prior, $Z_{\hat{\tau}} = 3$ since there are three valid configurations.

2.3.1 Message Passing Algorithms

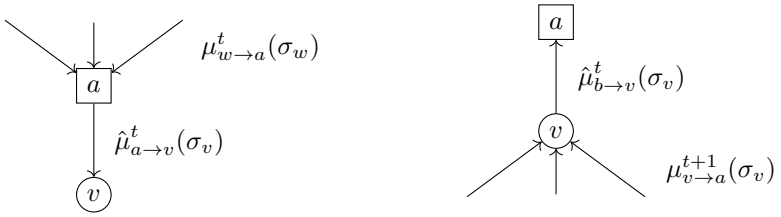
Message passing algorithms model the local interactions in a factor graph by assigning messages along the edges. In each step, these messages are updated for all edges using the former adjacent messages. This is repeated until (hopefully) the messages stabilize. BP has the algorithmic appeal that each of the update messages is only a local function in the graph, i.e. for each edge the update only has to consider adjacent previous messages, not the global state. In this section the most prominent example, Belief Propagation (BP) is introduce as well as a simplified version, Warning Propagation (WP).

2.4 Belief Propagation

Belief Propagation (BP) has been discovered in different fields independently [94]. It is known as Bethe-Peierls approximation in the field of statistical physics [17, 100] independently discovered by two different physicists, Bethe in 1935 and Peierls in 1936. Further in information theory it is known as the sum-product algorithm introduced by Gallager 1963 [54] and lastly under the name of Belief Propagation by Pearl in 1982 [99]. Belief Propagation is a powerful tool that estimates the marginals by iteratively updating of the factor graphs. Each edge $\{a, v\}$ comes with a pair of two messages, one from factor node to variable node $\hat{\mu}^t_{a\to v}$ and one from the variable node to the factor node $\mu^t_{v\to a}$. Intuitively, the message $\hat{\mu}^{t+1}_{a\to v}$ contains information about v's marginal distribution based on as neighbours excluding v. In this way, BP tracks local dependencies throughout the graph. We follow the BP-updates as given by [94], remember that $\propto$ hides the normalisation constant.

$$\mu^{t+1}_{v\to a}(\sigma_v) \propto \prod_{b\in\partial v\setminus\{a\}} \hat{\mu}^t_{b\to v}(\sigma_v) \tag{2.4.1}$$

$$\hat{\mu}^t_{a\to v}(\sigma_v) \propto \sum_{\sigma_{\partial a\setminus\{v\}}} \psi_{a,\hat{\tau}_a}(\sigma_{\partial a}) \prod_{w\in\partial a\setminus\{v\}} \mu^t_{w\to a}(\sigma_v) \tag{2.4.2}$$



(a) BP updates from factor to variable (b) BP updates from variable to factors

Figure 2.4: BP updates

Based on these messages we can give an estimate of the marginal distribution

$$\mu^t_v(\sigma_v) \propto \prod_{j\in\partial v} \hat{\mu}^{t+1}_{a\to v}(\sigma_v) \tag{2.4.3}$$

For the special case of acyclic factor graphs BP is exact:

Fact 2.4.1 ([94, Chapter 14]). *For a Gibbs measure μ represented by an acyclic Factor graph G the BP marginals as defined in* (2.4.3) *are exact, i.e.*

$$\lim_{t\to\infty} \mu_i^t(\sigma_i) = \mathbb{P}[\boldsymbol{\sigma}_i^* = \sigma_i].$$

The statement above does not hold for general graphs. BP might not even reach a fixed point and a fixed point is not necessarily correct. In principle, there could be a multitude of fixed points, where the result of BP heavily depends on the initialisation of the BP messages $\hat{\mu}^0_{a\to v}$ and $\mu^0_{v\to a}$. Generally, analysing BP rigorously is notoriously hard, however BP can be used to tackle difficult problems. For example in Section 3.4 a variant of BP is used to find an optimal solution that is verified afterwards.

For group testing the BP messages depend on the observed test results [8]. They come down to

$$\hat{\mu}^{t+1}_{a\to v}(1) = p_{1\hat{\tau}_a} \tag{2.4.4}$$

$$\hat{\mu}^{t+1}_{a\to v}(0) = p_{0\hat{\tau}_a} \prod_{w\in\partial a\setminus\{v\}} \mu^t_{w\to a}(0) + p_{1\hat{\tau}_a}\left(1 - \prod_{w\in\partial a\setminus\{v\}} \mu^t_{w\to a}(0)\right). \tag{2.4.5}$$

Warning Propagation is a boiled down form of BP where the messages carry the information of BP in an condensed form. In the case of *Constraint Satisfaction Problems* (CSPs) each factor node represents a constraint that is either satisfied or unsatisfied. Here BP can be simplified significantly as the messages collapse from a distribution to a single valued message. In Chapter 4 we use the technique of Warning Propagation to re-proof the k-XORSAT threshold. The update messages simplify to two different messages `f` and `u` and are given as [28, 34, 11]

$$\hat{\mathfrak{m}}_{v_j\to a_i} = \begin{cases} \mathtt{f} & \text{if } \exists a_h \in \partial v_j \setminus \{a_i\} : \mathfrak{m}_{a_h\to v_j} = \mathtt{f}, \\ \mathtt{u} & \text{otherwise,} \end{cases} \tag{2.4.6}$$

$$\hat{\mathfrak{m}}_{a_i\to v_j} = \begin{cases} \mathtt{f} & \text{if } \forall x_h \in \partial a_i \setminus \{v_j\} : \mathfrak{m}_{x_h\to a_i} = \mathtt{f}, \\ \mathtt{u} & \text{otherwise.} \end{cases}. \tag{2.4.7}$$

2.4.1 Bethe Free Entropy

Given a set of BP-messages μ the Bethe free entropy is a functional defined as follows [94, Chapter 14.2.4]:

$$\Phi(\mu) = \sum_{a\in F} F_a(\mu) + \sum_{v\in V} F_v(\mu) - \sum_{(a,v)\in E} F_{av}(\mu) \tag{2.4.8}$$

with (2.4.9)

$$F_v(\mu) = \ln\left(\sum_{\sigma_v} \prod_{b\in\partial v} \hat{\mu}^t_{b\to v}(\sigma_v)\right) \tag{2.4.10}$$

$$F_a(\mu) = \ln\left(\sum_{\sigma_{\partial a}} \psi_{a,\hat{\tau}_a}(\sigma_{\partial a}) \prod_{v\in\partial a} \mu^t_{v\to a}(\sigma_v)\right) \tag{2.4.11}$$

$$F_{av}(\mu) = \ln\left(\sum_{\sigma_v} \hat{\mu}^t_{b\to v}(\sigma_v)\mu^t_{w\to a}(\sigma_v)\right) \tag{2.4.12}$$

(2.4.13)

Further, if the marginal distribution for these messages is correct, the Bethe free entropy gives us [94]

$$\ln Z = \Phi(\mu).$$

The Bethe free entropy appears in Theorems 9 and 10 as part of the k-XORSAT threshold.

2.5 Genie Based Estimator

The genie based estimator is purely theoretical construct that is potentially more powerful than any possible estimator [1, 2]. It decode each position i of the ground truth given all information about the other dimensions excluding i. Since the genie based estimator is given more information to decode it is obvious that if it cannot succeed, no decoding algorithm can. The success of the genie based estimator is called the *local stability* property and characterizes the ability to discern between solutions with *high overlap* [33]. The idea of the genie based estimator can also be used to improve approximate solutions by repetitively update the estimates according to the genie [33, 65]. Under certain conditions this leads to an iterative erasure of mistakes.

The genie based estimator decodes each position $i \in [n]$ separately using a maximum a posteriori (MAP) decoder with access to an oracle that gives the true state of every other position. For a vector X let $X_{[i]}$ be X without X_i.

$$\left(\hat{\sigma}_{\text{gen}}(\hat{\tau}, \sigma)\right)_i = \arg\max_{\sigma_i} \mathbb{P}\left[\boldsymbol{\sigma}_i^* = \sigma_i \,\middle|\, \hat{\boldsymbol{\tau}} = \hat{\tau}, \boldsymbol{\sigma}_{[-i]}^* = \sigma_{[-i]}\right] \tag{2.5.1}$$

The genie estimator is not directly related to Belief Propagation, however, it coincides with the maximizer of the BP estimate for each variable if the second neighbourhood is fixed to the correct value. For each variable this results in a small acyclic graph containing exactly the second neighbourhood. Since BP is exact on acyclic graphs (Fact 2.4.1) the estimate for each variable is correct given the correct second neighbourhood states.

While BP is notoriously hard to analyse in most cases the genie estimator comes down to a local formulation. Since we assume that the second neighbourhood is decoded correctly decoding i is not depending on the estimate of the second neighbourhood but only on the true states. The only challenge left for the genie is to differentiate between the ground truth and configurations with a *high overlap* to the ground truth, i.e. configurations where only few indices are changed.

In Group Testing the genie based estimator can be used to lower bound the number of tests necessary to distinguish *high overlap* solutions. This technique is explicitly used in Section 3.5 for the proof of the lower bound Theorem 5 and in Section 3.4 as the local stability of the bound in Theorem 4. However, the idea of the genie based estimator is also used in decoding algorithms. Here, without access to an actual oracle, one can use the genie on an pre-emptive estimation with few errors to "tidy up" this pre-emptive estimation. This process of tidying up an pre-emptive solution is used for the algorithmic results Theorem 3 and Theorem 6.

Chapter 3

Group Testing

Group testing was first introduced by Dorfman [44] with the original motivation of large-scale testing for syphilis. It has been of continuing research interest since 1970 [70, 45, 15, 24, 8, 74] motivated by direct applications such as HIV testing in blood donations [48]. Of course, during the COVID-19 pandemic it gained further prominence [93]. However, the relevance exceeds the origin as there exist a vast variance of applications to areas beyond epidemiology [8, 27] e.g. multiple access communication [126] and data compression and storage [68].

3.1 Settings

In group testing we consider a large population of n individuals of which each a subset of individuals are infected. The general task is to infer $\boldsymbol{\sigma}^*$ by conducting m pooled tests $a_1, \dots, a_m$ which are actually positive if and only if at least one contained individual is infected, i.e. $\boldsymbol{\tau}'_a = \mathbb{1}\left\{\sum_{i \in \partial a} \boldsymbol{\sigma}^*_i \geq 1\right\}$. There are countless variations of the detailed settings of group testing, here we are briefly introducing the different aspects that are important later.

Binary Group Testing and Combinatorial vs. Probabilistic Group Testing Normally, we assumed that each individual is either infected or non infected and that the actual test result is only a binary information, which is the classical *binary* group testing setting. Further, the assumption that each individual is infected independently with a probability $\mathbb{P}[\boldsymbol{\sigma}^*_i = 1]$ is called the *probabilistic* setting. Another commonly studied prior is a fixed number of exactly k infected individuals. However, in most cases both settings can easily transferred into each other as described in Appendix to Chapter 1 of [8].

Sublinear vs. Linear Group Testing First, the probability $\mathbb{P}[\boldsymbol{\sigma}^*_i = 1]$ can either be constant as in the *linear* setting, or sublinear in n i.e. n^θ/n for $\theta \in (0, 1)$ which we call the *sublinear* or *sparse* case. The sublinear case is the dominant variant in literature [31, 25, 59] and especially interesting when studying beginning of epidemic process [123]. On the other hand, regarding applications to endemic or not transmitted diseases the assumption of a constant infection probability is more appropriate.

Noiseless vs. Noisy Group Testing The *noiseless* setting assumes that $\hat{\boldsymbol{\tau}}$ are precisely the *correct* results, i.e. that a test is observed positive iff one contained individual is infected. However, for testing for infections this is not realistic [63, 119]. In the *noisy* setting we assume that the actual test results are observed through a binary noisy channel $\boldsymbol{p} = (p_{00}, p_{01}, p_{10}, p_{11})$. An actually positive test (i.e. a test that contains at least one infected individual) is correctly observed positively with probability p_{11} and renders a false negative test result with probability p_{10}. Similarly, an actually negative tests is observed negatively with probability p_{00} and false positively p_{01}. Hence, the probability of an observed test result $\hat{\boldsymbol{\tau}}_a$ of a test a with actual test result $\boldsymbol{\tau}'_a$ is given as

$$\mathbb{P}[\tilde{\boldsymbol{\tau}}_a \,|\, \boldsymbol{\tau}'_a] = p_{\boldsymbol{\tau}'_a, \tilde{\boldsymbol{\tau}}_a}. \tag{3.1.1}$$

We require $\boldsymbol{p}$ to satisfy the natural properties $p_{01} + p_{00} = p_{11} + p_{10} = 1$ and $p_{11} > p_{01}$. Special cases include symmetric noise ($p_{01} = p_{10}$), the z-channel ($p_{00} = 1$) and the reverse z-channel ($p_{11} = 1$).

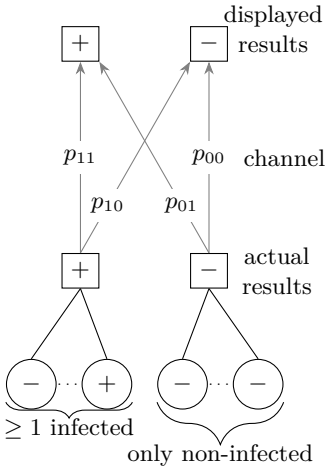


Figure 3.1: Noise channel.

Non Adaptive vs. Adaptive Group Testing There are two major scenarios for conducting the tests. They can either be conducted all at once in *non adaptive* group testing that has been extensively studied [25, 113, 112]. Before conducting the tests every test is fixed and cannot depend on any other test result. This can lead to faster available results with the expense of more required tests. In the non adaptive case we model the *test design* as a factor graph $G = (V, F, E)$ where each test $a \in F$ is connected to a variable $v \in F$ iff the corresponding individual participates in the test.

In *adaptive* group testing tests are conducted successively [45]. Since the choice of tests can be adjusted while testing this can lead to fewer necessary tests. We follow the definition of the included work [65]. We distinguish between our testing scheme $\boldsymbol{\mathcal{A}}$ that determines the next tests conducted based on the prior observed test results and the resulting test design $\boldsymbol{G}_{\boldsymbol{\mathcal{A}}}$. The resulting test graph $\boldsymbol{G}_{\boldsymbol{\mathcal{A}}}$ is a random variable where not only the test results depend on $\boldsymbol{\sigma}^*$ but also the conducted tests themselves, as they are allowed to depend on previous test results. Formally, an *adaptive test scheme* $\boldsymbol{\mathcal{A}} = (\boldsymbol{\mathcal{A}}_i)_{i\in[m]}$ is a sequence of sets forming a test design $G_{\boldsymbol{\mathcal{A}}} = (V_{\boldsymbol{\mathcal{A}}}, F_{\boldsymbol{\mathcal{A}}}, E_{\boldsymbol{\mathcal{A}}})$ with $F_{\boldsymbol{\mathcal{A}}} = \{a_1, \ldots, a_m\}$ such that for each $i \in [m]$, $\boldsymbol{\mathcal{A}}_i = \boldsymbol{\mathcal{A}}_{a_i} = \partial_{\boldsymbol{\mathcal{A}}} a_i$ is a (possibly random) function on the previous tests and their observed results. More precisely, $\boldsymbol{\mathcal{A}}_i = \boldsymbol{\mathcal{A}}_i((\boldsymbol{\mathcal{A}}_j, \hat{\boldsymbol{\tau}}_j)_{j\in[i-1]}) \subseteq [n]$, with $\hat{\boldsymbol{\tau}}_j$ as the observed test result of $\boldsymbol{\mathcal{A}}_j$.

Approximate vs. Exact Recovery The requirements for the precision of the estimate can also differ. Usually we want an estimator $\hat{\sigma}$ to be *exact* w.h.p., i.e. $\mathbb{P}[\boldsymbol{\sigma}^* = \hat{\sigma}] = 1 - o_n(1)$. We always refer to exact recovery when not further specified. However, one can also loosen this requirement to save some tests. Thus, for *approximate recovery* we only require the result to have a high overlap with the ground truth. More precisely, given that we have k infected individuals, $\mathbb{P}[\|\boldsymbol{\sigma}^* - \hat{\sigma}\|_1 \leq \varepsilon k] > 1 - \varepsilon$.

Further Variations On top of the aforementioned settings, there is a huge diversity on group testing variations that we will not discuss further. These include settings where the tests do not only transmit binary information but *quantitative* information about the number of contained tests [47, 21, 50]. Of course, these variations can be (and are) combined with the already mentioned settings, leading to a unmanageable number of settings in group testing.

3.2 Prior Work

An excellent overview of the work in group testing prior to 2019 in can be found in [8]. Studies on the group testing problem go back to the 1960' where noiseless linear group testing has been studied by Ungar motivated by the goal to identify syphilis infected soldiers using as few as possible expensive tests. Ungar showed that in the noiseless case, even if adaptiveness is allowed, one cannot do better than individual testing the fraction of infected individual exceeds $\alpha = \frac{1}{2}(3 - \sqrt{5})$ [120]. Far later an important result of Aldridge shows that noiseless non-adaptive group testing does not perform better than individual

testing [5]. For α below the threshold above Aldridge proposed an adaptive testing scheme [6] based on binary splitting [70]. However, the correct bounds in this regime are still unknown.

The optimality of individual testing for the noiseless linear regime led to research on non-adaptive group testing to focus on the sublinear regime for a considerable amount of time. This regime is now understood completely [31], following a considerable amount of prior work [7, 112, 74, 30]. During this time, two different test designs (Bernoulli and constant column) and various elementary algorithms have been proposed [23]. Among these, the DD algorithm on the constant column design achieved the best performance, but without matching the known information-theoretic bound for all θ [30] Late a more sophisticated test design based on spatial coupling was proposed [31, 32]. This contribution proved that the algorithm SPIV on a spatially coupled test design matches the improved information theoretic lower bound on the number of tests necessary. Therefore the non adaptive noiseless sublinear group testing is solved optimally both for exact and approximate recovery.

For noisy sublinear group testing the noisy version of the greedy DD algorithm has been studied for the Bernoulli and the Constant Column test design [44, 113]. On the latter it achieved the best performance for exact recovery previous to our result [59]. As for the information-theoretic side, most of the prior results have been focused on restrictive noise models like symmetric noise ($p_{11} = p_{00}$). For symmetric noise Chen and Scarlett pinned down the threshold of tests needed for exact recovery on the Bernoulli and Constant Column test design. Similar to the noiseless case, the Constant Column test design performs better than the Bernoulli Design. In yet another work, Scarlett and Cevher [112] have obtained precise bounds for approximate recovery, again under the assumption of a symmetric noise channel. Neither of the mentioned results on the symmetric channel deal with the question of an efficient algorithm. However, applying out results we re-obtain the aforementioned sharp bounds.

If we turn to noisy linear group testing, rigorous results are rather sparse. The only rigorous results in this setting is a lower bound on the necessary tests only for the special case of symmetric noise that turns out to be off by a factor of $(1 - 2p_{01})^{-1}$ [111]. This work also contains a three-stage test scheme for symmetric noise similar to ours and analyses it only for the sublinear regime.

3.3 Overlap

In the following, we discuss results for group testing in multiple settings including both upper and lower bounds. Even though each of these results is based on different proof methods there are some basic properties that extend to all settings. A working test design (for exact recovery) is able to differentiate the true solution from any other configuration. More precisely, it has to separate the ground truth from both solutions with low overlap and high overlap.

Low overlap For the low overlap solutions, we basically have to distinguish between the correct solution and any random solution. For understanding this condition we are going to borrow some intuition from coding theory. We have to distinguish $\binom{n}{k}$ different possible configurations of the individuals on the "input" side. On the other hand, we receive m binary bits of information through the tests. However, since each test is not observed directly, but through a noise channel, each test does not contain a "full" bit of information. Intuitively, the amount of information that can be send through a binary channel per bit is exactly determined by the Shannon capacity of a channel $c_{\mathrm{Sh}}(\boldsymbol{p})$. Even though this does not suffice as a proof, this intuition leads to the tight bound for approximate group testing Theorems 1 and 2 and to one of the conditions of Theorems 3 and 4. Remembering Fig. 1.1, excluding low overlap solutions translates to the condition, that the probability mass of all solution clusters far away from $\boldsymbol{\sigma}^*$ (area of the “small bubbles”) together vanish.

High overlap On the other side, we also have to distinguish between high overlap configurations - those, who have a diminishing but existing error compared to the ground truth. Assume that we are given the correct states of each individual except one. In a working test design it has to be possible to decode this last individual. Therefore, we have to be able to distinguish between different but very similar configurations - in this thought experiment exactly configurations where exactly one individual is missing. The genie estimator of Section 2.5 formalizes this intuition further by repeating this step for every single individual.

To understand this notion a bit better, we have to consider what the information about any other individual gives us exactly. For example, consider an individual x with positive and negative tests in its' neighbourhood ∂x. Clearly, if a test $a \in \partial x$ contains a *different* infected individual, then we know that a should be a positive test, and we know that another infected individual is responsible for a being positive. Consequently, a does not carry any information about x any more, since the state of x does not in any way effect a. Hence, we call such tests *uninformative* and further, tests that do not contain a different infected individual *informative*. We will call g_x the number of informative or *good* tests for an individual x and g_x^+, g_x^- the number of positively or negative displayed informative tests.

Intuitively for non adaptive group testing, decoding x under these assumptions can only depend on g_x, g_x^+, g_x^-. More precisely, there has to exist a threshold-function $\mathfrak{z} : \mathbb{N} \to \mathbb{N}$ that assigns every number of good tests g_x a threshold $\mathfrak{z}(g_x)$ s.t. x is diagnosed as infected if $\mathfrak{z}(g_x) < g_x^+$. This is in fact precisely what the genie estimator does which is proven for a special case in Lemma 4.3 of Appendix C. Conversely, if we can find (many) infected and uninfected individuals, for which there exists no such threshold, we cannot hope to decode these individuals correctly.

This is a little bit more involved for adaptive group testing, since the tests (and the number of good test) are not independent. However, the notion of good tests for this case is still a useful tool for proving Theorem 7. Remembering Fig. 1.1, excluding high overlap solutions w.h.p. translates the probability mass of the solution cluster including $\boldsymbol{\sigma}^*$ focusing solely on $\boldsymbol{\sigma}^*$.

3.4 Sublinear Regime

The results of this section are taken from

Noisy group testing via spatial coupling

by Amin Coja-Oghlan, Max Hahn-Klimroth, Lukas Hintze, Dominik Kaaser, Lena Krieg, Maurice Rolvien and Olga Scheftelowitsch published in *Combinatorics, Probability and Computing*.

This work studies the case of sublinear noisy group testing and solves the problem of approximate and exact recovery almost completely. More precisely, for the case of approximate recovery a lower bound on necessary tests is established as well as an efficient algorithm that matches this lower bound. For the case of exact recovery the paper provides a lower bound for the Constant Column test design and as well as an efficient algorithm that matches this lower bound.

Define for two probabilities q and p the Kullback-Leibler divergence, also called relative entropy, as $\mathrm{D}_{\mathrm{KL}}(p \parallel q) = p\ln(p/q) + (1-p)\ln((1-p)/(1-q))$. With the channel capacity of the noise channel $\boldsymbol{p}$ given as [33, Eq. (1.3)]

$$\phi = \phi(\boldsymbol{p}) = \frac{h(p_{00}) - h(p_{10})}{p_{00} - p_{10}}\ , \qquad c_{\mathrm{Sh}} = c_{\mathrm{Sh}}(\boldsymbol{p}) = \frac{1}{\mathrm{D}_{\mathrm{KL}}(p_{10} \parallel (1-\tanh(\phi/2))/2)} \tag{3.4.1}$$

the first two results pin down the threshold of tests needed for approximate recovery as

$$m_{\mathtt{SPARC}} = c_{\mathrm{Sh}} \ln \binom{n}{k}.$$

Intuitively, this threshold provides that the test design has to "transmit" $\ln\binom{n}{k}$ "bits" of information about $\boldsymbol{\sigma}^*$, where each test individually can at most transmit $1/c_{\mathrm{Sh}}$ bits of information [116].

Theorem 1 ([33, Theorem 1.1])**.** *For any $\boldsymbol{p}$, $0 < \theta < 1$ and $\varepsilon > 0$ there exists $n_0 = n_0(\boldsymbol{p}, \theta, \varepsilon)$ such that for every $n > n_0$ there exist a randomised test design $\mathbf{G}_{\mathrm{sc}}$ with $m \le (1+\varepsilon)m_{\mathtt{SPARC}}(n, k, \boldsymbol{p})$ tests and a deterministic polynomial time inference algorithm* SPARC *such that*

$$\mathbb{P}[\|\mathtt{SPARC}(\mathbf{G}_{\mathrm{sc}}, \hat{\boldsymbol{\tau}}_{\mathbf{G}_{\mathrm{sc}}}) - \boldsymbol{\sigma}^*\|_1 < \varepsilon k] > 1 - \varepsilon. \tag{3.4.2}$$

Theorem 2 ([33, Theorem 1.2])**.** *For any $\boldsymbol{p}$, $0 < \theta < 1$ and $\varepsilon > 0$ there exist $\delta = \delta(\boldsymbol{p}, \theta, \varepsilon) > 0$ and $n_0 = n_0(\boldsymbol{p}, \theta, \varepsilon)$ such that for all $n > n_0$, all adaptive test designs with $m \le (1-\varepsilon)m_{\mathtt{SPARC}}(n, k, \boldsymbol{p})$ tests in total and any function $\mathcal{A} : \{0,1\}^m \to \{0,1\}^n$ we have*

$$\mathbb{P}[\|\mathcal{A}(\hat{\boldsymbol{\tau}}) - \boldsymbol{\sigma}^*\|_1 < \delta k] < 1 - \delta. \tag{3.4.3}$$

Note, that Theorem 2 does bound the success probability of any algorithm away from 1, however it does not provide a sharp threshold, since it does not exclude the existence of an algorithm with constant success probability. The question, whether any algorithm fails w.h.p. (i.e. the success probability tends to zero) remains open.

For exact recovery, the threshold itself has a more complicated form and comes as a non trivial optimisation problem. More precisely, additionally to the condition of being able to discern between configurations that are far apart from each other (i.e. have a low overlap) as in approximate recovery, we need to ensure that we are able to differentiate between similar solutions. In the constant column design this enforces a competing restriction on the degrees of the infected individuals.

Following [33, Eq. (1.6 – 1.11)], for $c, d > 0$ and $\theta \in (0,1)$ let

$$\mathcal{Y}(c,d,\theta) = \{y \in [0,1] : cd(1-\theta)\,\mathrm{D_{KL}}\left(y \parallel \exp(-d)\right) < \theta\}\,. \tag{3.4.4}$$

This set is a non-empty interval, because $y \mapsto \mathrm{D_{KL}}\left(y \parallel \exp(-d)\right)$ is convex and $y = \exp(-d) \in \mathcal{Y}(c,d,\theta)$. Let

$$c_{\mathrm{ex},0}(d,\theta) = \begin{cases} \inf\left\{c > 0 : \inf_{y \in \mathcal{Y}(c,d,\theta)} cd(1-\theta)\left(\mathrm{D_{KL}}\left(y \parallel \exp(-d)\right) + y\,\mathrm{D_{KL}}\left(p_{01} \parallel p_{11}\right)\right) \geq \theta\right\} & \text{if } p_{11} < 1, \\ \inf\{c > 0 : 0 \notin \mathcal{Y}(c,d,\theta)\} & \text{otherwise.} \end{cases} \tag{3.4.5}$$

If $p_{11} = 1$ let $\mathfrak{z}(y) = 1$ for all $y \in \mathcal{Y}(c,d,\theta)$. Further, if $p_{11} < 1$ then the function $z \mapsto \mathrm{D_{KL}}\left(z \parallel p_{11}\right)$ is strictly decreasing on $[p_{01}, p_{11}]$; therefore, for any $c > c_{\mathrm{ex},0}(d,\theta)$ and $y \in \mathcal{Y}(c,d,\theta)$ there exists a unique $\mathfrak{z}(y) = \mathfrak{z}_{c,d,\theta}(y) \in [p_{01}, p_{11}]$ such that

$$cd(1-\theta)\left(\mathrm{D_{KL}}\left(y \parallel \exp(-d)\right) + y\,\mathrm{D_{KL}}\left(\mathfrak{z}(y) \parallel p_{11}\right)\right) = \theta. \tag{3.4.6}$$

In either case set

$$c_{\mathrm{ex},1}(d,\theta) = \begin{cases} \inf\Big\{c > c_{\mathrm{ex},0}(d,\theta) : \\ \qquad \inf_{y \in \mathcal{Y}(c,d,\theta)} cd(1-\theta)\left(\mathrm{D_{KL}}\left(y \parallel \exp(-d)\right) + y\,\mathrm{D_{KL}}\left(\mathfrak{z}(y) \parallel p_{01}\right)\right) \geq 1\Big\} & \text{if } p_{01} > 0, \\ c_{\mathrm{ex},0}(d,\theta) & \text{otherwise.} \end{cases} \tag{3.4.7}$$

Finally, define

$$c_{\mathrm{ex},2}(d) = 1\big/\left(h(p_{00}\exp(-d) + p_{10}(1-\exp(-d))) - \exp(-d)h(p_{00}) - (1-\exp(-d))h(p_{10})\right), \tag{3.4.8}$$

$$c_{\mathrm{ex}}(\theta) = \inf_{d>0} \max\{c_{\mathrm{ex},1}(d,\theta), c_{\mathrm{ex},2}(d)\}, \tag{3.4.9}$$

$$m_{\mathtt{SPEX}}(n,k,\boldsymbol{p}) = c_{\mathrm{ex}}(\theta)k\ln(n/k).$$

The following pair of results provide the exact threshold for exact recovery in the constant column test design.

Theorem 3 ([33, Theorem 1.3])**.** *For any $\boldsymbol{p}$, $0 < \theta < 1$ and $\varepsilon > 0$ there exists $n_0 = n_0(\boldsymbol{p},\theta,\varepsilon)$ such that for every $n > n_0$ there exist a randomised test design $\mathbf{G}_{\mathrm{sc}}$ with $m \leq (1+\varepsilon)m_{\mathtt{SPEX}}(n,k,\boldsymbol{p})$ tests and a deterministic polynomial time inference algorithm* SPEX *such that*

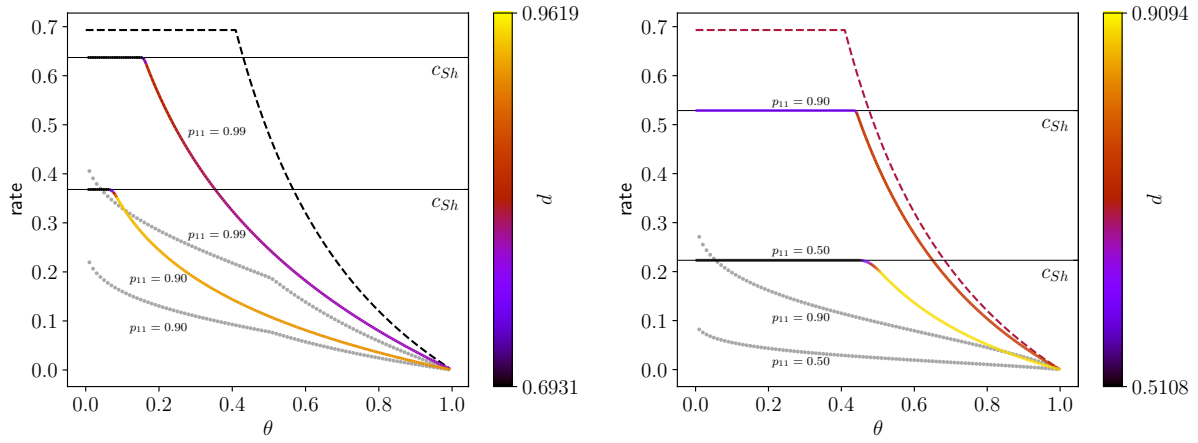
$$\mathbb{P}[\mathtt{SPEX}(\mathbf{G}_{\mathrm{sc}}, \hat{\boldsymbol{\tau}}_{\mathbf{G}_{\mathrm{sc}}}) = \boldsymbol{\sigma}^*] > 1 - \varepsilon. \tag{3.4.10}$$

Theorem 4 ([33, Theorem 1.4])**.** *For any $\boldsymbol{p}$, $0 < \theta < 1$ and $\varepsilon > 0$ there exists $n_0 = n_0(\boldsymbol{p},\theta,\varepsilon)$ such that for every $n > n_0$ and all $m \leq (1-\varepsilon)m_{\mathtt{SPEX}}(n,k,\boldsymbol{p})$, $\Delta > 0$ and any $\mathcal{A}_{\mathbf{G}_{\mathrm{cc}}} : \{0,1\}^m \to \{0,1\}^n$ we have*

$$\mathbb{P}[\mathcal{A}_{\mathbf{G}_{\mathrm{cc}}}(\hat{\boldsymbol{\tau}}_{\boldsymbol{G}}) = \boldsymbol{\sigma}^*] < \varepsilon. \tag{3.4.11}$$

3.4.1 Rates

To visualize the difference that our algorithm SPEX achieves compared top the most efficient prior algorithm noisy DD, the rates achieved by both for different level of noise is displayed in Fig. 3.2. This figure displays the amount of information transferred per necessary test, higher rates means that more information is transmittable and less tests needed. The difference between noisy DDand SPEX is quiet blatant, as SPEX achieves constantly higher rates and for some choices of θ achieves even higher rates than noisy DD for way higher noise levels. This means, that using as many tests as noisy DD, SPEX manages to diagnose the infected patients for noise level that are at least up to 10 times higher Fig. 3.2(a). On another note, in Fig. 3.2 we also depict the optimal d for the spatial coupled test design used by SPEX. This parameter essentially determines the degree of the variable nodes of $\mathbf{G}_{\text{sc}}$. The optimal parameter is chosen as a non trivial optimisation problem (3.4.8). Fig. 3.2 shows, that there is a conflict between the necessary conditions of (3.4.8) leading to the different values of d visible in the curvature.



(a) Rates on the binary symmetric channel for $p_{00} = p_{11} = 0.99$ and $p_{00} = p_{11} = 0.9$.

(b) Rates on the Z-channel ($p_{00} = 1$) with $p_{11} = 0.9$ and $p_{11} = 0.5$.

Figure 3.2: Information rates on different channels in nats. The horizontal axis displays the infection density parameter $0 < \theta < 1$. The colour indicates the optimal value of d for a given θ. [33, Figure 1]

To visualize the difference both between exact recovery and approximate recovery as well as the gain of SPEX and SPARC compared to the next best efficient test design and recovery algorithm, noisy DD of [59], Fig. 3.2 shows the rates of both for different noise levels. For $m = ck\ln(n/k)$ the rate is defined as $\ln\binom{n}{k}/m = 1/c$ and measures of the amount of information transmitted per test. Here, a higher rate is better, since then fewer tests are necessary to transmit the necessary information. Note, that for certain θ, SPEX outperforms noisy DDso much, that it needs less tests even with ten times higher noise levels.

3.4.2 Spatially Coupled Test Design

Using ideas from random graphs to tackle inference problem has lead to many achievements, among capacity achieving linear codes via spatially coupled low-density parity check ('LDPC') codes [85, 84, 60] . The combination of a global spatial arrangement with local randomness enables efficient decoding through BP. This idea has extended to other inference problems as well, with *compressed sensing*, solving an underdetermined linear system restricted by a sparsitity contraint [42, 43, 81, 83], as the most prominent example. Here, a variant of BP called Approximate Message Passing combined with a spatially spatially coupled graph matches the information -theoretic lower bound [127].

The present test design is conceptionally identical to the test design of [31] used for noiseless group testing. It combines the concepts of a constant column test design with an overlying spatial structure. Locally, the test design "look" and behaves like a normal constant column test design, whereas the individuals and tests ar partitioned into compartments that are ordered in a circle. This order will act as a guide for the main step SPARC.

Following [33, Sec. 2.3] the test design partitions the individuals and the tests in $\ell = \Theta(\sqrt{\ln n})$ equally sized compartments $V[1], \dots, V[\ell], F[1], \dots, F[\ell]$. The individual and test compartments are both ordered into a ring structure as visualized in Fig. 3.3. Further, we let the *sliding window* be $s = \lceil \ln \ln n \rceil$. Each variable randomly joins $\Delta = cd \ln(n/k) = \Theta(\ln n)$ tests evenly spread over the next s test compartments. More precisely, each variable $x \in V[i]$ joins Δ/s randomly chosen tests of test compartments $i + j - 1$ for $1 \leq j \leq s$ Additionally, we add another test compartment $F[0]$ that contains the necessary amount of tests to decode the first s individual compartments $V[1], \dots, V[s]$ using the noisy `DD` algorithm.

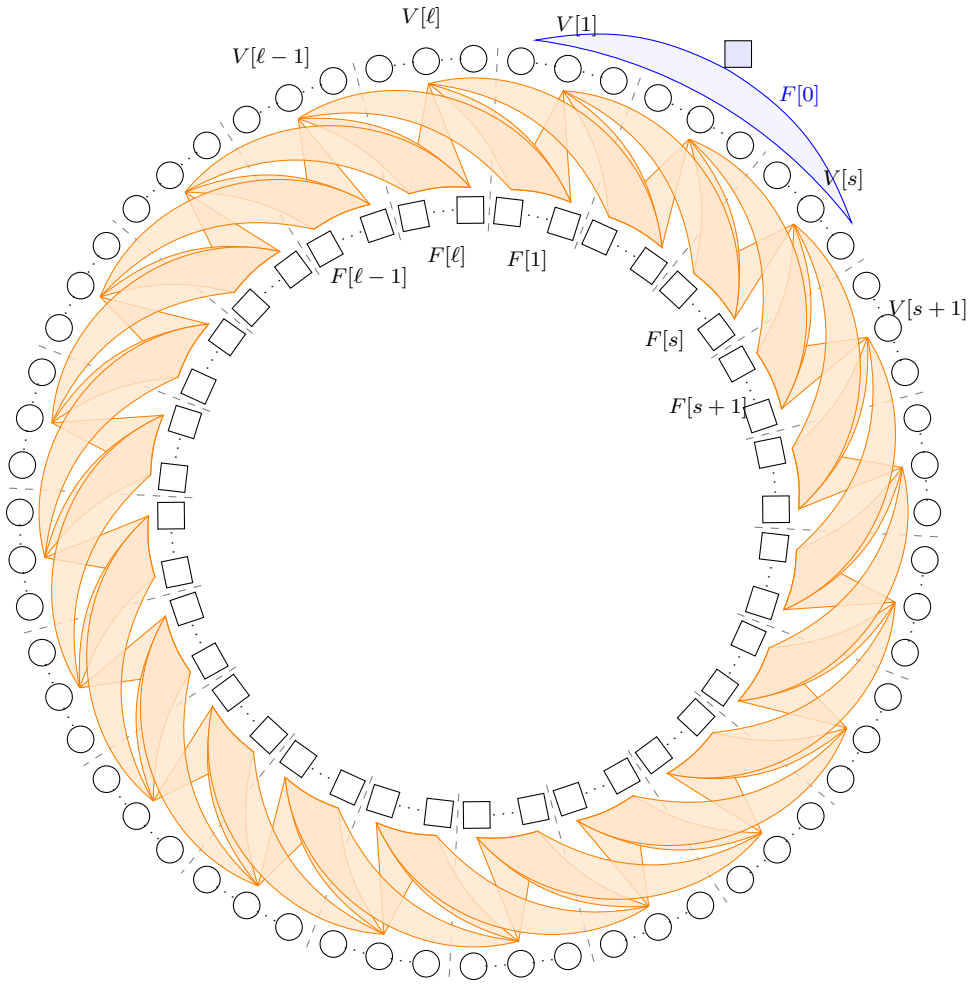


Figure 3.3: Sketch of the spatially coupled group testing design.

The decoding strategy of `SPARC` works as follows. First, it starts by decoding the seed compartment correctly using `DD` and the tests of $F[0]$. Then `SPARC` iterates through the individual compartments $V[1], \dots, V[\ell]$. In each iteration we consider a compartment $V[i]$ under the assumption that every individual of $V[1], \dots, V[i-1]$ is diagnosed correctly. Note, that since we decoded the first s departments correctly via noisy `DD` we can indeed assume this when starting from compartment $s+1$. Now consider a individual $v \in V[i]$ and informative adjacent tests of test compartments $\{F[i+j-1] : 1 \leq j \leq s\}$. Intuitively, tests of compartment $F[i-s+1]$ contain more information about v, since they have fewer undiagnosed adjacent individuals than tests of compartment $F[i]$. Hence, we should weight these tests differently depending on which compartment they belong to. Let $\boldsymbol{W}_{i,j}(\tau)$ be the set of all informative tests $a \in F[i+j-1]$. We assign weights w_j^+, w_j^- to each positive and negative informative test that contains v. We will threshold this sums of weights s.t. we identify every individual as healthy as long as these weights behaves as expected from a negative individual and positive otherwise. The sum of these

weights will be the indicator for diagnosing x and are defined as [33, Eq. (2.30)].

$$\boldsymbol{W}_v^+(\tilde{\boldsymbol{\tau}}) = \sum_{j=1}^{s} w_j^+ \left|\boldsymbol{W}_{v,j}^+(\tilde{\boldsymbol{\tau}})\right|, \qquad \boldsymbol{W}_v^-(\tilde{\boldsymbol{\tau}}) = \sum_{j=1}^{s} w_j^- \left|\boldsymbol{W}_{v,j}^-(\tilde{\boldsymbol{\tau}})\right| \tag{3.4.12}$$

To diagnose v we compare this value to the expectation conditioned on v's infection state $W^+ = \mathbb{E}\left[\boldsymbol{W}_v^+(\tilde{\boldsymbol{\tau}}) \mid \sigma_v = 1\right]$ and $W^- = \mathbb{E}\left[\boldsymbol{W}_v^+(\tilde{\boldsymbol{\tau}}) \mid \sigma_v = 0\right]$. This turns out to be [33, Eq. (2.32)]

$$W^+ = p_{11}\Delta \sum_{j=1}^{s} \exp(d(j-s)/s) w_j^+, \quad W^- = \begin{cases} p_{10}\Delta \sum_{j=1}^{s} \exp(d(j-s)/s) w_j^- & \text{if } p_{10} > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3.4.13}$$

With optimally chosen weights, this indeed leads to a working algorithm. If we on the other hand consider a paced version of BP, we receive exactly the optimal weights for `SPARC`. The optimal weights are [33, Eq. (2.29)]

$$w_j^+ = \ln \frac{p_{11}}{p_{11} + (p_{01} - p_{11})\exp(-dj/s)} \geq 0, \quad w_j^- = -\ln \frac{p_{10}}{p_{10} + (p_{00} - p_{10})\exp(-dj/s)} \geq 0. \tag{3.4.14}$$

3.4.3 `SPARC` Algorithm

The `SPARC` algorithm now implements precisely the decoding strategy as described above by iterating over the compartments and calculating the summed weights for every individual.

Data: $\boldsymbol{G}$, $\hat{\boldsymbol{\tau}}$
Result: an estimate of $\boldsymbol{\sigma}^*$
Let $(\tau_x)_{x \in V[1] \cup \cdots \cup V[s]} \in \{0,1\}^{V[1] \cup \cdots \cup V[s]}$ be the result of applying `DD` to $V[1] \cup \cdots \cup V[s]$ and $F[0]$;
Set $\tau_x = *$ for all individuals $x \in V \setminus (V[1] \cup \cdots \cup V[s])$;
for $i = s+1, \ldots, \ell$ **do**
  for $x \in V[i]$ **do**
    if $x \notin V^+[i]$ *or* $\boldsymbol{W}_x^+(\tau) < (1-\zeta)W^+$ *or* $\boldsymbol{W}_x^-(\tau) > (1+\zeta)W^-$ **then**
      $\tau_x = 0$ `// classify as uninfected`
    else
      $\tau_x = 1$ `// classify as infected`
return τ

Algorithm 1: the `SPARC` Algorhtm [33, Algorithm 1 and 2]

3.4.4 Local Stability

`SPARC` returns an approximate solution of the ground truth where at most $o(k)$ individuals are misclassified, so the ground truth has a high overlap to the $\hat{\boldsymbol{\sigma}}_{\texttt{SPOG}}$ w.h.p. The approach of `SPEX` is now to 'tidy' up the solution by repeatedly running local thresholding steps. Conceptionally, we run a genie based estimator started on the pre-estimate instead of the ground truth. However, a necessary condition for this to work is the *local stability* of the solution. Already mentioned in Section 3.3 this property ensures, that given a correct second neighbourhood solution, each individual can be correctly classified w.h.p. In other words, the genie based estimator has to be correct on the test design, else we cannot expect to correctly achieve exact recovery. In fact, Theorem 4 provides, that exact recovery is impossible on the constant column test design when local stability is not satisfied.

The precise conditions can be derived as follows. For a constant column design, each individual has degree Δ. If we have a threshold z_y that separates infected and non infected individuals w.h.p., we have to exclude the existence of a pair v^-, v^+ such that

- v^-, v^+ have both $y\Delta$ informative tests
- v^- has at least $z_y y\Delta$ positive informative tests
- v^+ has at most $z_y y\Delta$ positive informative tests.

If this happens, we cannot hope to separate v^- from v^+. This is also visualised in Fig. 3.4

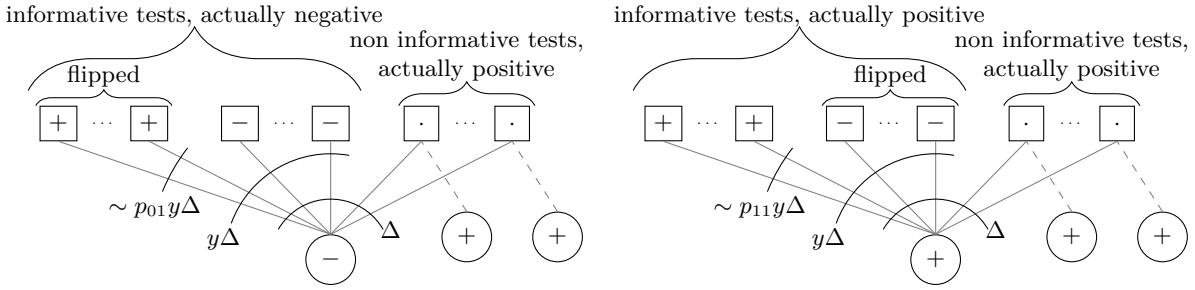


Figure 3.4: Local stability.

3.4.5 SPEX

Using the results of the optimisation that lead to the threshold Theorem 3, SPEX does tidies up the approximate solution of SPARC and irons out every mistake w.h.p. Note, that we use the threshold function $\mathfrak{z}(y)$ as defined in (3.4.6).

Data: $\mathbf{G}_{\mathrm{sc}}$, $\hat{\boldsymbol{\tau}}$
Result: an estimate of $\boldsymbol{\sigma}^*$
Let $\tau^{(1)}$ be the output of $\mathtt{SPARC}(\mathbf{G}_{\mathrm{sc}}, \hat{\boldsymbol{\tau}})$;
for $i = 1, \dots, \lceil \ln n \rceil$ **do**
    For all $x \in V[s+1] \cup \cdots \cup V[\ell]$ calculate

$$Y_x(\tau^{(i)}) = \sum_{a \in \partial x \setminus F[0]} \mathbb{1}\left\{\forall y \in \partial a \setminus \{x\} : \tau_y^{(i)} = 0\right\},$$

$$Z_x(\tau^{(i)}) = \sum_{a \in \partial x \setminus F[0] : \hat{\boldsymbol{\tau}}_a = 1} \mathbb{1}\left\{\forall y \in \partial a \setminus \{x\} : \tau_y^{(i)} = 0\right\};$$

    Let $\tau_x^{(i+1)} =$

$$\begin{cases} \tau_x^{(i)} & \text{if } x \in V[1] \cup \cdots \cup V[s], \\ \mathbb{1}\left\{Y_x(\tau^{(i)})/\Delta \in \mathcal{I} \text{ and } Z_x(\tau^{(i)})/\Delta > \mathcal{Z}(Y_x(\tau^{(i)})/\Delta)\right\} & \text{otherwise} \end{cases};$$

return $\tau^{(\lceil \ln n \rceil)}$

Algorithm 2: The SPEX algorithm [33, Algorithm 3].

3.5 Linear Regime

The results of this section from the paper

Noisy Linear Group Testing: Exact Thresholds and Efficient Algorithms

by Lukas Hintze, Lena Krieg, Olga Scheftelowitsch and Haodong Zhu (accepted for COLT) provides a complete solution for exact recovery in linear noisy group testing. Both for non adaptive as well as adaptive group testing we provide sharp thresholds for the number of tests as well as efficient algorithms that achieve exact recovery and match these threshold. Therefore, we solve one of the most interesting models for group testing completely, providing efficient algorithms and optimal sharp bounds for general noise.

For non adaptive group testing the threshold comes in a form of an optimization problem [65, Eq. (2.2)]:

$$m_{\mathrm{na}} = m_{\mathrm{na}}(\alpha, \boldsymbol{p}) = \min_{\Gamma \in \mathbb{N}^+} \frac{n \ln n}{-\Gamma \cdot \ln\left(1 - (1-\alpha)^{\Gamma-1} \cdot \left(1 - \mathrm{e}^{-\beta(\boldsymbol{p})}\right)\right)} \tag{3.5.1}$$

When $(1-\varepsilon)m_{\mathrm{na}}$ tests are used, any non-adaptive algorithm will fail with high probability:

Theorem 5 ([65, Theorem 1]). *For any valid positive noisy channel $\boldsymbol{p}$, $\alpha > 0$ and $\varepsilon > 0$, there exists some $n_0 = n_0(\boldsymbol{p}, \alpha, \varepsilon)$ such that for every $n > n_0$, all test designs G with $m < (1-\varepsilon)m_{\mathrm{na}}(\alpha, \boldsymbol{p})$ tests and every estimation function $f_G : \{0,1\}^m \to \{0,1\}^n$:*

$$\mathbb{P}[f_G(\hat{\boldsymbol{\tau}}_G) = \boldsymbol{\sigma}^*] \leq \varepsilon.$$

Furthermore, the next theorem provides that the non-adaptive algorithm `SPOG` given in Algorithm 3 recovers $\boldsymbol{\sigma}^*$ with high probability using $(1+\varepsilon)m_{\mathrm{na}}$ tests.

Theorem 6 ([65, Theorem 2]). *For any valid positive noisy channel $\boldsymbol{p}$, $\alpha > 0$ and $\varepsilon > 0$, there exists some $n_0 = n_0(\boldsymbol{p}, \alpha, \varepsilon)$ such that for every $n > n_0$, there is a randomized test design $\boldsymbol{G}$ using $m \leq (1+\varepsilon)m_{\mathrm{na}}(\alpha, \boldsymbol{p})$ tests w.h.p. and a deterministic polynomial time algorithm `SPOG` such that*

$$\mathbb{P}[\mathit{SPOG}(\boldsymbol{G}, \hat{\boldsymbol{\tau}}_{\boldsymbol{G}}) = \boldsymbol{\sigma}^*] \geq 1 - \varepsilon.$$

For adaptive group testing, the following results state that

$$m_{\mathrm{ad}} = m_{\mathrm{ad}}(\alpha, \boldsymbol{p}) = \frac{\alpha}{\mathrm{D_{KL}}\left(p_{11} \parallel p_{01}\right)} \cdot n \ln n$$

is the threshold for necessary tests. When at most $(1-\varepsilon)m_{\mathrm{ad}}$ tests are used, any adaptive test scheme and estimator fail with high probability:

Theorem 7 ([65, Theorem 3]). *For any valid positive noisy channel $\boldsymbol{p}$, $\alpha > 0$ and $\varepsilon > 0$, there exists some $n_0 = n_0(\boldsymbol{p}, \alpha, \varepsilon)$ such that for every $n > n_0$, any adaptive test scheme $\boldsymbol{\mathcal{A}}$ using $m_{\boldsymbol{\mathcal{A}}} < (1-\varepsilon)m_{\mathrm{ad}}(\alpha, \boldsymbol{p})$, and any estimation algorithm $f_{G_{\boldsymbol{\mathcal{A}}}} : \{0,1\}^{m_{\boldsymbol{\mathcal{A}}}} \to \{0,1\}^n$:*

$$\mathbb{P}[f_{G_{\boldsymbol{\mathcal{A}}}}(\hat{\boldsymbol{\tau}}_{\boldsymbol{\mathcal{A}}}) = \boldsymbol{\sigma}^*] \leq \varepsilon. \tag{3.5.2}$$

However, we introduce the non-adaptive efficient algorithm `PRESTO` that succeeds with high probability using at most $(1+\varepsilon)m_{\mathrm{ad}}$ tests.

Theorem 8 ([65, Theorem 4]). *For any valid positive noisy channel $\boldsymbol{p}$, $\alpha > 0$ and $\varepsilon > 0$, there exists some $n_0 = n_0(\boldsymbol{p}, \alpha, \varepsilon)$ such that for every $n > n_0$, the three-stage adaptive test scheme `PRESTO` uses at most $m \leq (1+\varepsilon)m_{\mathrm{ad}}(\alpha, \boldsymbol{p})$ w.h.p. such that*

$$\mathbb{P}[\mathit{PRESTO}(\hat{\boldsymbol{\tau}}_{\mathit{PRESTO}}) = \boldsymbol{\sigma}^*] \geq 1 - \varepsilon. \tag{3.5.3}$$

3.5.1 Impossibility Non Adaptive

The proof of the lower bound has three main parts. First, we use the fact that the genie estimator is at least as good as any other estimator on any given test design. Hence, when the genie estimator fails w.h.p., so does any other estimator as well. First we characterize the genie estimator as a threshold function given the number of good tests and positive good tests.

Next, we transform the arbitrary original test design carefully into a test design that can only perform better by providing even more information to the decoding algorithm. We are using this modified test design to identify a relatively large set of individuals that are hard to diagnose. Finally, we show that the genie estimator fails w.h.p. already on this identified set of individuals, concluding our proof.

The Genie Estimator is characterizable as a threshold function given the neighbourhood of each individual i according to the following lemma.

Lemma 3.5.1 ([65, Lemma 4.3]). *Let G be a test design, $C = \ln(\frac{p_{00}}{p_{10}})/\ln(\frac{p_{11}p_{00}}{p_{01}p_{10}})$, and $\kappa = \kappa(\alpha, \boldsymbol{p}) = \ln(\frac{\alpha}{1-\alpha})/\ln(\frac{p_{01}p_{10}}{p_{11}p_{00}})$. Then*

$$\hat{\sigma}_{\mathrm{gen}}{}^{G}(i) = \begin{cases} 0 & \text{if } g_i(\boldsymbol{\sigma}^*) = 0 \text{ and } \alpha \leq \frac{1}{2}, \text{ or } g_i^+(\boldsymbol{\sigma}^*) \leq C g_i(\boldsymbol{\sigma}^*) + \kappa, \\ 1 & \text{otherwise.} \end{cases}$$

Adjusted Test Design The adjusted test design is defined as follows.

Definition 3.5.2 ([65, Definition 4.5])**.** Given test design G, we the *modified test design* G_η is constructed as follows:

- remove tests in $L = \left\{a \in F \mid |\partial_G a| \geq \ln^2(n)\right\}$ to obtain design G',
- remove individuals in $J = \left\{i \in [n] \mid \left|\partial_{G_{\eta,L}} i\right| > \ln^4(n) - \lceil \eta \ln(n) \rceil\right\}$ to obtain design G'', and
- add $\lfloor \eta \ln(n) \rfloor$ individual tests for each individual i with $\eta > 0$ to obtain design G_η. ♦

The following lemma provides, that the adjusted test design contains only more information and that diagnosing the individuals only becomes easier through these adjustments.

Lemma 3.5.3 (modified test setup is easier, [65, Lemma 4.6])**.** *Let $\boldsymbol{\sigma}^*[\check{n}]$ be the infection statuses for individuals in $[\check{n}]$ and $\hat{\sigma}[\check{n}]$ be the prediction of the infection statuses for individuals in $[\check{n}]$ based on estimator $\hat{\sigma}$. Then*

$$\mathbb{P}\left[\boldsymbol{\sigma}^*[\check{n}] = \hat{\sigma}_{\mathrm{gen}}{}^{G_\eta}[\check{n}]\right] \geq \mathbb{P}\left[\boldsymbol{\sigma}^*[\check{n}] = \hat{\sigma}_{\mathrm{MAP}}{}^{G_\eta}[\check{n}]\right] \geq \mathbb{P}\left[\boldsymbol{\sigma}^* = \hat{\sigma}_{\mathrm{MAP}}{}^{G'}\right] \geq \mathbb{P}\left[\boldsymbol{\sigma}^* = \hat{\sigma}_{\mathrm{MAP}}{}^{G}\right] - o(1)\,. \tag{3.5.4}$$

Distant Set of Individuals Now in this modified test design, we find a relatively large set of individuals whose tests are independent. This is ensured by not including individuals whose tests intersect,

Lemma 3.5.4 (probability of misclassification [65, Lemma 4.8])**.** *Let $i \in [\check{n}]$, and let $0 < \delta < 1 - p_{10} - C$. Then for sufficiently large n, the probability that i is misclassified by the genie estimator is bounded from below as*

$$\mathbb{P}\left[\hat{\sigma}_{\mathrm{gen}}{}^{G_\eta}(i) \neq \boldsymbol{\sigma}^*(i) \,\middle|\, \boldsymbol{\sigma}^*\right] \geq \exp\left(-g_i(\boldsymbol{\sigma}^*, G_\eta)\, \mathrm{D_{KL}}\left(C - \delta \parallel p_{11}\right)\right). \tag{3.5.5}$$

All that remains is to find a distant set D such that the sum of the lower bounds given by Lemma 3.5.4 over all $i \in D$ is large for most $\boldsymbol{\sigma}^*$. The following lemma achieves this.

Lemma 3.5.5 ([65, Lemma 4.9])**.** *For any $\varepsilon > 0$, there exist δ and $\eta > 0$ such that for any $\delta'' > 0$, there is a $n_0(\delta'')$ so that for all $n \geq n_0(\delta'')$, any test design G on n individuals using at most $(1-\varepsilon)m_{\mathrm{na}}$ tests, there is a distant set $D \subseteq [\check{n}]$ so that*

$$\mathbb{P}\left[\sum_{i \in D} \exp\left(-\,\mathrm{D_{KL}}\left(C - \delta \parallel p_{11}\right) g_i(\boldsymbol{\sigma}^*, G_\eta)\right) > \frac{1}{2\delta''}\right] \geq 1 - 4\delta''. \tag{3.5.6}$$

Putting together these pieces concludes the proof of Theorem 5.

Proof of Theorem 5 following [65, Sec. 4.1]. For sufficient large n, the $o(1)$ term in Lemma 3.5.3 is at most $\varepsilon/2$. It now suffices to show that the probability of the genie estimator being correct on G_η is at most $\varepsilon/2$ for sufficiently large n. To bound the probability of the genie estimator being correct on $[\check{n}]$, combining the above for a set of distant individuals $D \subseteq [\check{n}]$ gives us,

$$\mathbb{P}\left[\boldsymbol{\sigma}^*[\check{n}] = \hat{\sigma}_{\mathrm{gen}}{}^{G_\eta}[\check{n}] \,\middle|\, \boldsymbol{\sigma}^*\right] \leq \left(\sum_{i \in D} \exp\left(-g_i(\boldsymbol{\sigma}^*, G_\eta)\, \mathrm{D_{KL}}\left(C - \delta \parallel p_{11}\right)\right)\right)^{-1}. \tag{3.5.7}$$

Now write $\boldsymbol{U}_D$ for this upper bound. By Lemma 3.5.5, for any δ'' and sufficiently small δ'', we can choose D so that $\mathbb{P}[\boldsymbol{U}_D \leq 2\delta''] \geq 1 - 4\delta''$. And as trivially $\mathbb{P}[\boldsymbol{\sigma}^*[\check{n}] = \hat{\sigma}_{\mathrm{gen}}{}^{G_\eta}[\check{n}] \mid \boldsymbol{\sigma}^*] \leq 1$, we have

$$\mathbb{P}\left[\boldsymbol{\sigma}^*[\check{n}] = \hat{\sigma}_{\mathrm{gen}}{}^{G_\eta}[\check{n}]\right] = \mathbb{E}\left[\mathbb{P}\left[\boldsymbol{\sigma}^*[\check{n}] = \hat{\sigma}_{\mathrm{gen}}{}^{G_\eta}[\check{n}] \,\middle|\, \boldsymbol{\sigma}^*\right]\right] \leq \mathbb{E}[\min\{1, \boldsymbol{U}_D\}] \tag{3.5.8}$$

$$\leq \mathbb{E}[\min\{1, \boldsymbol{U}_D\} \,|\, \boldsymbol{U}_D \leq 2\delta''] + \mathbb{P}[\boldsymbol{U}_D > 2\delta''] = 2\delta'' + 4\delta'', \tag{3.5.9}$$

which yields the theorem with $\delta'' = \varepsilon/12$. □

3.5.2 Impossibility Adaptive

The proof for the lower bound of the adaptive part consists of three main steps following [65, Sec. 5.1]. First, we add a few individual tests, this only improves the estimator. Further, in this adjusted test scheme we define so called *typical* infection vectors, that include the ground truth w.h.p. Finally, we show that the a posteriori possibility of *any* given typical infection vector tends to zero.

Let $\boldsymbol{\mathcal{A}}$ be a test scheme according with fewer than m_{ad} tests. Further let

$$\eta = \frac{\varepsilon}{2} \cdot \frac{m_{\mathrm{ad}}}{n \ln n} = \frac{\varepsilon}{2} \cdot \frac{\alpha}{\mathrm{D}_{\mathrm{KL}}\left(p_{11} \parallel p_{01}\right)}.$$

We obtain a modified test scheme $\boldsymbol{\mathcal{A}}'$ from $\boldsymbol{\mathcal{A}}$ by adding a small number of $\lfloor \eta \ln(n) \rfloor$ single tests for each individual.

Typical infection vectors First, let us characterise define the notion of a typical infection vector as follows.

Definition 3.5.6 (typical infection vector [65, Definition 5.3])**.** Let $\varepsilon' > 0$. For any infection vector $\sigma \in \{0,1\}^n$, test design G, and displayed test results $\hat{\tau}$, define the set of infected individuals with a ε'-typical ratio of good tests displaying positively as

$$\mathcal{J}_{\sigma}^{\varepsilon'} = \mathcal{J}_{\sigma,G,\hat{\tau}}^{\varepsilon'} = \left\{ i \in \mathcal{I}_{\sigma} \mid \left| g_i^+(\sigma) - p_{11} g_i(\sigma) \right| \leq \varepsilon' g_i(\sigma) \right\}.$$

An infection vector σ is *ε'-typical* for G and $\hat{\tau}$ if $|\mathcal{J}_{\sigma,G,\hat{\tau}}^{\varepsilon'} - \alpha n| \leq \varepsilon' n$; Let $\mathcal{C}^{\varepsilon'}(G,\hat{\tau})$ be the set of these σ. ♦

The following lemma provides that indeed, the ground truth is a typical infection vector w.h.p.

Lemma 3.5.7 ([65, Lemma 5.4])**.** *For any $\varepsilon' > 0$, $\boldsymbol{\sigma}^*$ is ε'-typical for $G_{\boldsymbol{\mathcal{A}}'}$ and $\hat{\boldsymbol{\tau}}_{\boldsymbol{\mathcal{A}}'}$ w.h.p.:*

$$\mathbb{P}[\boldsymbol{\sigma}^* \notin \mathcal{C}^{\varepsilon'}(G_{\boldsymbol{\mathcal{A}}'}, \hat{\boldsymbol{\tau}}_{\boldsymbol{\mathcal{A}}'})] = o(1).$$

Posteriori Probability of typical infection vectors The following lemma provides, that th posterior probability of every infection vector tends to zero if we are not supplied with enough tests.

Lemma 3.5.8 ([65, Lemma 5.6])**.** *For all sufficiently small $\varepsilon' > 0$, any test design G and observed test results $\hat{\tau}$, and any ε'-typical infection vector $\sigma \in \mathcal{C}^{\varepsilon'}(G,\hat{\tau})$, $\mathbb{P}[\boldsymbol{\sigma}^* = \sigma \mid G_{\boldsymbol{\mathcal{A}}'} = G, \hat{\boldsymbol{\tau}}_{\boldsymbol{\mathcal{A}}'} = \hat{\tau}] = o(1)$.*

Combining this statement with the fact that the correct infection vector is typical w.h.p. leads to the following proof of Theorem 7.

Proof of Theorem 7 following [65, Sec. 5.1]. For any f, any $\varepsilon' > 0$, let $\boldsymbol{f} = f_{G_{\boldsymbol{\mathcal{A}}'}}(\hat{\boldsymbol{\tau}}_{\boldsymbol{\mathcal{A}}'})$ and $\boldsymbol{\mathcal{C}}^{\varepsilon'} = \mathcal{C}^{\varepsilon'}(G_{\boldsymbol{\mathcal{A}}'}, \hat{\boldsymbol{\tau}}_{\boldsymbol{\mathcal{A}}'})$. Then

$$\begin{aligned}\mathbb{P}[\boldsymbol{\sigma}^* = \boldsymbol{f}] &= \mathbb{P}\left[\boldsymbol{\sigma}^* = \boldsymbol{f} \wedge \boldsymbol{f} \in \boldsymbol{\mathcal{C}}^{\varepsilon'}\right] + \mathbb{P}\left[\boldsymbol{\sigma}^* = \boldsymbol{f} \wedge \boldsymbol{f} \notin \boldsymbol{\mathcal{C}}^{\varepsilon'}\right] \\ &\leq \mathbb{P}\left[\boldsymbol{\sigma}^* = \boldsymbol{f} \,\middle|\, \boldsymbol{f} \in \boldsymbol{\mathcal{C}}^{\varepsilon'}\right] + \mathbb{P}\left[\boldsymbol{\sigma}^* \notin \boldsymbol{\mathcal{C}}^{\varepsilon'}\right].\end{aligned}$$

Since $\mathbb{P}[\boldsymbol{\sigma}^* \notin \boldsymbol{\mathcal{C}}^{\varepsilon'}] = o(1)$ for any $\varepsilon' > 0$ by Lemma 3.5.7, if the first term vanishes, the claim follows. When choose $\varepsilon' > 0$ sufficiently small, by Lemma 3.5.8 it follows that:

$$\begin{aligned}&\mathbb{P}\left[\boldsymbol{\sigma}^* = \boldsymbol{f} \,\middle|\, \boldsymbol{f} \in \boldsymbol{\mathcal{C}}^{\varepsilon'}\right] \\ =& \sum_{\substack{G,\hat{\tau}, \\ \sigma \in \mathcal{C}^{\varepsilon'}(G,\hat{\tau})}} \mathbb{P}\left[\boldsymbol{\sigma}^* = \sigma \,\middle|\, \sigma \in \boldsymbol{\mathcal{C}}^{\varepsilon'}, \boldsymbol{f} = \sigma, G_{\boldsymbol{\mathcal{A}}'} = G, \hat{\boldsymbol{\tau}}_{\boldsymbol{\mathcal{A}}'} = \hat{\tau}\right] \cdot \mathbb{P}\left[\boldsymbol{f} = \sigma, G_{\boldsymbol{\mathcal{A}}'} = G, \hat{\boldsymbol{\tau}}_{\boldsymbol{\mathcal{A}}'} = \hat{\tau} \,\middle|\, \boldsymbol{f} \in \boldsymbol{\mathcal{C}}^{\varepsilon'}\right] \\ =& o(1) \cdot \sum_{\sigma} \mathbb{P}\left[\boldsymbol{f} = \sigma \,\middle|\, \boldsymbol{f} \in \boldsymbol{\mathcal{C}}^{\varepsilon'}\right] = o(1).\end{aligned}$$

□

3.5.3 SPOG Algorithm

SPOG has three different types of tests that are all conducted in parallel. First, we conduct a relatively small number of individual tests F_1 for each individual. We utilize this small number of tests to define a pre-estimation that has a small number of mistakes. Then, we add a number of random tests as well as a small number of extra test to ensure that each individual is contained in at least enough tests. SPOG is going to use the pre estimate and these group tests to run a pseudo genie estimation for each individual. To exclude the possibility of a misclassification in the end, we have to choose the number of tests as well as the degrees such that first, the pre estimate has a small number of tests. Second the distribution of the number of *good* tests for each individual has to suffice that, even with the false positive and negative observed test results, the pseudo genie does not make a single mistake whatsoever.

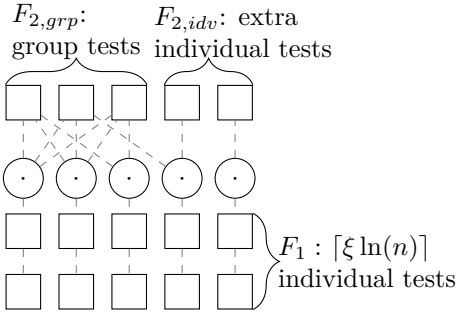


Figure 3.5: Illustration of $\boldsymbol{G}_{\texttt{SPOG}}$. Circles represent individuals and squares tests [65, Figure 2].

Data: Instance G of $\boldsymbol{G}_{\texttt{SPOG}}$, $\hat{\boldsymbol{\tau}}_G \in \{0,1\}^n$
for $i \in [n]$ **do**
  $\hat{\boldsymbol{\sigma}}^{(1)}(i) \leftarrow \mathbb{1}\left[\left|F_1^+(i)\right| \geq C \cdot |F_1(i)|\right]$
for $i \in [n]$ **do**
  $S \leftarrow \emptyset$, $D_i \leftarrow \emptyset$
  for $a \in F_2 \cap \partial i$ **do**
    if $\partial a \setminus \{i\} \cap S \neq \emptyset$ **then**
      $D_i \leftarrow D_i \cup \{a\}$, $S \leftarrow S \cup (\partial a \setminus \{i\})$
  $P_i \leftarrow \{a \in D_i \mid \forall j \in \partial a \setminus \{i\} : \hat{\boldsymbol{\sigma}}^{(1)}(i) = 0\}$
  $\hat{\boldsymbol{\sigma}}_{\texttt{SPOG}}(i) \leftarrow \mathbb{1}\left[|a \in P_i \cap \partial i \mid \hat{\boldsymbol{\tau}}(a) = 1| \geq C \cdot |P_i|\right]$
return $\hat{\boldsymbol{\sigma}}_{\texttt{SPOG}}$

Algorithm 3: The SPOG algorithm [65, Algorithm 1]

3.5.4 Sublinear Mismatched SPOG

In the adaptive testing scheme, we will iteratively extract a group of individuals with a sublinear number of infected individuals contained. To diagnose this group, we use a subroutine for sublinear noisy group testing. However, since we separate this group by the means of possibly noisy tests, the number of infected individuals can actual vary, and the correct prior in the subroutine might differ from the expected one. For this purpose, we also show that for suitable parameters, SPOG is correct for sub-constant α, even if we only know an upper bound on the correct value of α.

Proposition 3.5.9 ([65, Proposition 4.18])**.** *For any valid noisy channel $\boldsymbol{p}$, and any $\hat{\theta} \in (0,1)$ and $\varepsilon > 0$, there is a randomized test design G using $m \leq \varepsilon n \ln n$ tests w.h.p. so that as long as $\alpha \leq n^{-\hat{\theta}}$,*

$$\mathbb{P}[\textit{SPOG}(G, \hat{\boldsymbol{\tau}}_G) = \boldsymbol{\sigma}^*] \geq 1 - \varepsilon.$$

3.5.5 PRESTO Algorithm

For the adaptive testing, we use a test scheme that works in three rounds. First, similar to SPOG we conduct a relatively small amount of individual tests to get a pre estimate for each individual. We use

this pre estimate to partition the individuals into a set with a sublinear number of infected individuals and a set with a high amount of infected individuals. The latter receives a large number of individual tests. These tests are again used to distil a set of individuals where each individual is infected w.h.p. and a set with a sub linear number of infected individuals. The sets with a sublinear number of infected individuals are then diagnosed using the sublinear variant of SPOG using a negligible amount of tests.

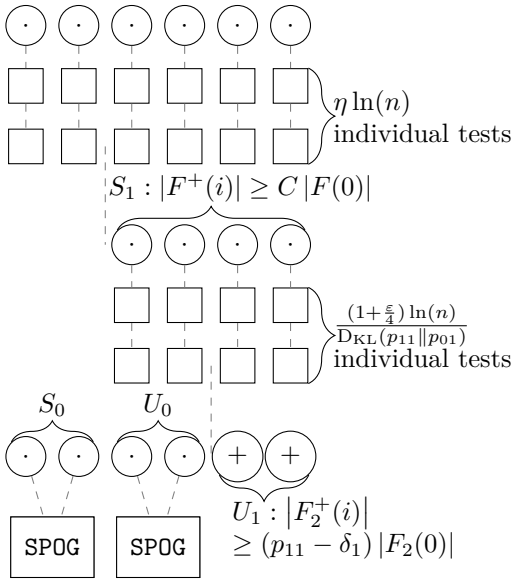


Figure 3.6: An illustration of the adaptive test scheme of PRESTO. The circles represent the individuals and the squares represent tests [65, Figure 3].

The parameters $\eta > 0$ and $\delta_1 \in (0, p_{11} - p_{10})$ are chosen so that

$$\left(1 + \frac{\varepsilon}{4}\right) \frac{\mathrm{D_{KL}}\left(p_{11} - \delta_1 \parallel p_{01}\right)}{\mathrm{D_{KL}}\left(p_{11} \parallel p_{01}\right)} > 1, \tag{3.5.10}$$

and

$$\eta\, \mathrm{D_{KL}}\left(C \parallel p_{01}\right) < \min\left\{1, \left(1 + \frac{\varepsilon}{4}\right) \frac{\mathrm{D_{KL}}\left(p_{11} - \delta_1 \parallel p_{11}\right)}{\mathrm{D_{KL}}\left(p_{11} \parallel p_{01}\right)}\right\}. \tag{3.5.11}$$

Parameters: $\varepsilon > 0$, $\delta_1 \in (0, p_{11} - p_{01}), \eta > 0$ satisfying Eqs. (3.5.10) and (3.5.11).

1 Conduct $\lceil \eta \ln(n) \rceil$ individual tests $F_1(i)$ for each $i \in [n]$;

2 Let $F_1^+(i)$ be the positive ones.

3 $S_1 \leftarrow \left(i \in [n] \mid \left|F_1^+(i)\right| > C\,|F_1(i)|\right)$

4 $S_0 \leftarrow [n] \setminus S_1$

5 Conduct $\lceil \left(1 + \frac{\varepsilon}{4}\right) \frac{\ln(n)}{\mathrm{D_{KL}}(p_{11}\|p_{01})} \rceil$ individual tests $F_2(i)$ for each $i \in S_1$;

6 let $F_2^+(i)$ be the positive ones.

7 $U_1 \leftarrow \left(i \in S_1 \mid \left|F_2^+(i)\right| > (p_{11} - \delta_1)\,|F_2(i)|\right)$

8 $U_0 \leftarrow S_1 \setminus U_1$

9 $\hat{\boldsymbol{\sigma}}^S_{\texttt{SPOG}} \leftarrow$ result of SPOG run on S_0 as described in Proposition 3.5.9, with $\hat{\theta} = \hat{\theta}^S = \frac{\eta}{2}\,\mathrm{D_{KL}}\left(C \parallel p_{11}\right)$.

10 $\hat{\boldsymbol{\sigma}}^U_{\texttt{SPOG}} \leftarrow$ result of SPOG sun on U_0 as described in Proposition 3.5.9, with $\hat{\theta} = \hat{\theta}^U = \frac{1}{2}\left(\left(1 + \frac{\varepsilon}{4}\right) \frac{\mathrm{D_{KL}}(p_{11} - \delta_1 \| p_{11})}{\mathrm{D_{KL}}(p_{11} \| p_{01})} - \eta\,\mathrm{D_{KL}}\left(C \parallel p_{01}\right)\right)$.

11 **return** $\hat{\boldsymbol{\sigma}}_{\texttt{PRESTO}} = \left(\mathbb{1}\left[i \in U_1 \cup \mathcal{I}_{\hat{\boldsymbol{\sigma}}^S_{\texttt{SPOG}}} \cup \mathcal{I}_{\hat{\boldsymbol{\sigma}}^U_{\texttt{SPOG}}}\right]\right)_{i \in [n]}$

Algorithm 4: PRESTO[65, Algorithm 2]

Chapter 4

The XORSAT Threshold

One of the most famous NP hard problem is for an arbitrary SAT formula the question if this formula is satisfiable [37]. Even though SAT is NP hard, SAT instances that are based on real world problems tend to be solved easy by modern SAT solvers [89]. On the other side, random SAT instances without any structure or real world interpretation, are notoriously hard to solve efficiently by SAT solvers. This is one of many reasons why random SAT or related areas are of high interest for research. Cuckoo-hashing [40], a hashing algorithm with a worst case guarantees on search time, works until the k-XORSAT satisfiability threshold.

XORSAT has another important connection to the field of coding theory, where the codewords of Low-density parity check ('LDPC') codes are the solutions of a random k-XORSAT formula. Moreover, LDPC codes actually achieve channel capacity for a particular distribution of XORSAT formulas and decoding a message is efficiently possibly using Belief Propagation [84, 86].

The 3-XORSAT threshold was first proven by Dubois [46] stating that their proof is extendable relatively easily to the general k-XORSAT threshold. However, what was conjectured to be an easy adjustment turned out to be far more complicated and was first proven just over ten years later [101] in a very lengthy and complex proof that relied on computer assistance. Later a still complicated proof that abstains from computer assistance was published [11] based on coupling arguments.

The results of this section contribute a short, self contained prove for the k-XORSAT threshold, inspired by physicists techniques. First, we characterise random XORSAT formulas quantitatively using the physics inspired approach of Warning Propagation as introduced in Section 2.4 and carrying out what physicists call a 'quenched' analysis argument. We use these quantitative characterisation to carefully carry out a moment computation (called annealed analysis). Because we focus on solutions with the correct 'quenched' properties, the annealed analysis is tight and elegant.

4.1 Model

We follow the definitions of [34]. Let $\boldsymbol{F} = \boldsymbol{F}_k(n,m) = \wedge_{i=1}^{m} \oplus_{j=1}^{k} l_{i,j}$ be a random k-XORSAT instance over n Boolean variables $x_1, \dots, x_n$ and m random XOR clauses of length k. Each clause draws its variables independently at random out of all $\binom{n}{k}$ possible combinations of k variables and attaches a negation to each variable independently with probability 1/2.

There are two further natural representations and perspective of a XORSAT instance. First, we can translate the formula directly into a linear system of $\mathbb{F}_2$, where each XOR-clause gives us exactly one constraint. More precisely, a XOR clause c_i with variables $\ell_{i,1}\oplus, \dots, \oplus\ell_{i,k}$ states that the number of true entries for the literals is odd, hence let σ be an assignment and let $x_{i,j}$ be the variable of literal $\ell_{i,j}$, this leads to the constraint that

$$\sum_{j=1}^{k} \mathbb{1}\{\ell_{i,j} \text{ is negated}\} - \sigma_i = 1 \mod 2$$

If we move the negations to the right side of the equation, this is equal to

$$\sum_{j=1}^{k} \sigma_{x_{i,j}} = \mathbb{1}\{k + \text{the number of negations in } \ell_1, \dots, \ell_k \text{ is odd}\}$$

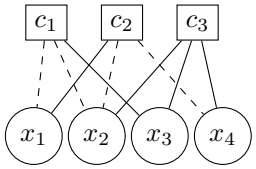
XOR-formula	Matrix-equation	Factorgraph
$x_1 \oplus x_2 \oplus \neg x_3$ $\wedge \neg x_1 \oplus x_2 \oplus x_4$ $\wedge \neg x_2 \oplus \neg x_3 \oplus \neg x_4$	$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$	

Table 4.1: Example of a 3-XOR formula and the corresponding matrix as well as the factor graph. A solid edges indicates that a variable appears negated in the connected clause.

Let y_i be the right side of the equation above, then negating each variable in a clause independently equals choosing y_i independently. Hence, this directly induces a matrix $\boldsymbol{A}$ where each column represents a variable, each row represents a clause and has exactly k non-zero entries. Thus, with $\boldsymbol{A}_{x,i} = \mathbb{1}\{x \in \mathrm{Var}[c_i]\}$ the system of equations is given as

$$\boldsymbol{A}\sigma = \boldsymbol{y}$$

Since $\boldsymbol{y}$ is independent of $\boldsymbol{A}$, the random XOR-formula $\boldsymbol{F}$ is satisfiable w.h.p. iff $\boldsymbol{A}$ has full row rank m w.h.p. More precisely, when $\boldsymbol{A}$ has full row rank, i.e. $\mathrm{rk}_{\mathbb{F}}(\boldsymbol{A})$ the system $\boldsymbol{A}x = \boldsymbol{y}$ has a solution for all $\boldsymbol{y}$, since else their exists two contradicting derived constraints. Conversely, if $\mathrm{rk}_{\mathbb{F}}(\boldsymbol{A}) < m - \Omega(n)$ there exists at least $\Omega(n)$ constraints that are already implied. Each of these constraints, since they are already implied by the other $m - \Omega(n)$ constraints, will lead to an unsatisfiable equation if the corresponding entry of $\boldsymbol{y}$ is does not coincide with the implied value. Since $\boldsymbol{y}$ is drawn independently of $\boldsymbol{A}$ and each entry of $\boldsymbol{y}$ is drawn by a coin flip, the probability of $\boldsymbol{A}x = \boldsymbol{y}$ is unsatisfiable tends to one.

An example for a 3-XORSAT formula and the associated matrix is depicted in Section 4.1

The representation as a random matrix over $\mathbb{F}_2$ can be directly generalized into sparse to a linear system over $\mathbb{F}_2$ inducing a random matrix over $\mathbb{F}_2$ where each column represents a variable and each row a clause of the formula [34]. Therefore, let q be a prime power. Let further $\mathfrak{A} = (\mathfrak{A}_{ij})_{i,j\geq 1}$ be an infinite matrix with non zero entries $\mathfrak{A}_{ij} \in \mathbb{F}_q \setminus \{0\}$. Further, choose a sequence $(\boldsymbol{e}_i)_{i\geq 1}$ of independent uniformly random subsets of $[n]$ of size $|e_i| = k$. Define the random $m \times n$-matrix $\boldsymbol{A} = \boldsymbol{A}(k, m, n, q, \mathfrak{A})$ over $\mathbb{F}_q$ by letting

$$\boldsymbol{A}_{ij} = \mathfrak{A}_{ij}\mathbb{1}\{j \in \boldsymbol{e}_i\} \qquad (i \in [m],\, j \in [n]). \tag{4.1.1}$$

This is also visualised in Section 4.1. For $q = 2$ we obtain the matrix induced by a k-XORSAT formula.

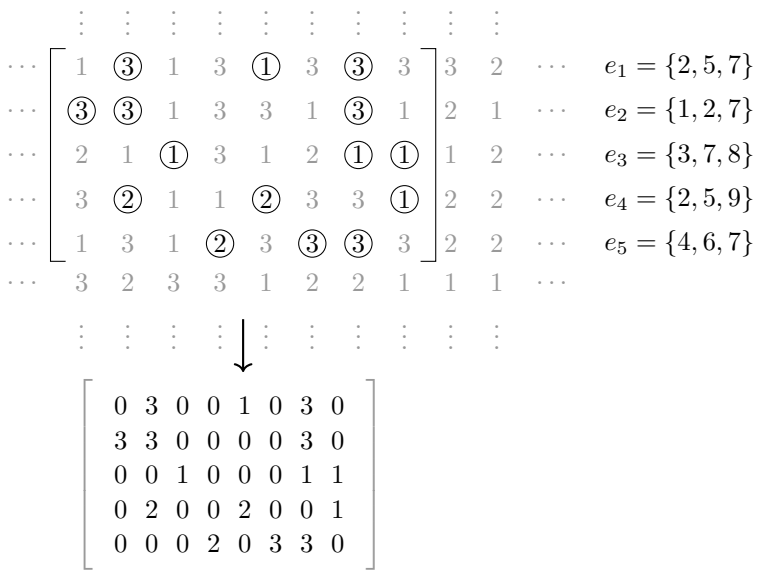


Figure 4.1: Visualisation of creation of $\boldsymbol{A}$ through $\mathfrak{A}$ using e

4.2 Contribution

The results of this section is from the paper

The k-XORSAT threshold revisited

by Amin Coja-Oghlan, Mihyun Kang, Lena Krieg and Maurice Rolvien published in the *Electronic Journal of Combinatorics* and contains two main contributions. First, it contains an elegant and short proof for the k-XORSAT threshold that relies on a combinatoric understanding and characterization of the structure of a random k-XORSAT instance and the contained variables. Second, it contains a generalised theorem that extends the k-XORSAT threshold to spars random matrices over $\mathbb{F}_q$.

Theorem 9 ([34, Theorem 1.1]). *For $k \geq 3$ and $d > 0$ let*

$$\Phi_{d,k}(\alpha) = \exp\left(-d\alpha^{k-1}\right) + d\alpha^{k-1} - \frac{d(k-1)}{k}\alpha^k - \frac{d}{k} \qquad \text{and} \tag{4.2.1}$$

$$d_k = \sup\left\{d > 0 : \max_{\alpha\in[0,1]} \Phi_{d,k}(\alpha) = 1 - d/k\right\}. \tag{4.2.2}$$

For any $\varepsilon > 0$ w.h.p. the random k-XORSAT formula $\boldsymbol{F}$ is

1. *satisfiable if $m \leq (1-\varepsilon)d_k n/k$,*
2. *unsatisfiable if $m \geq (1+\varepsilon)d_k n/k$.*

The proof of Theorem 9 carries over to the generalization of k-XORSAT formulas as introduced in Section 4.1 as follows.

Theorem 10 ([34, Theorem 1.2]). *For any $k \geq 3$, any prime power $q \geq 2$ and any infinite matrix $\mathfrak{A}$ composed of non-zero elements of $\mathbb{F}_q$ the following is true. Let d_k be the threshold from (4.2.1). Then for any $\varepsilon > 0$,*

1. *if $m \leq (1-\varepsilon)d_k n/k$, then $\boldsymbol{A}$ has full row rank w.h.p.*
2. *if $m \geq (1+\varepsilon)d_k n/k$, then $\boldsymbol{A}$ fails to have full row rank w.h.p.*

The Bethe free entropy is visualises for $k = 3$ in Fig. 4.2 and $k = 4$ in Section 4.2. The black curve indicates the Bethe free entropy for exactly the threshold, i.e. the point where the maxima of $\Phi_{d,k}(\alpha)$. For $d < d_k$ the Φ reaches its maximum near 1 on the left whereas for $d < d_k$ it is maximised by $\alpha = 0$.

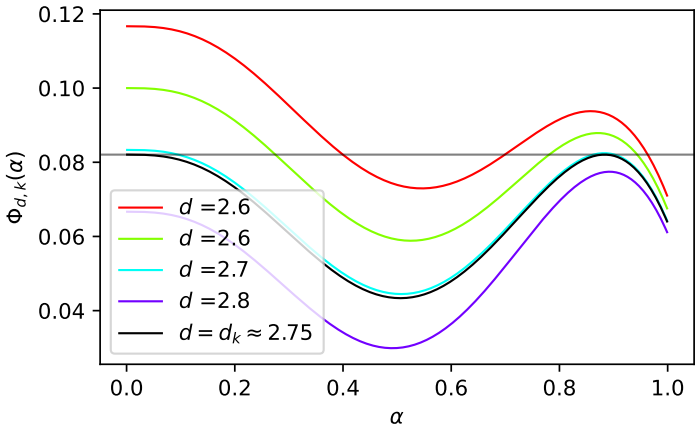


Figure 4.2: $\Phi_{d,3}$ from (4.2.1) for $k = 3$.

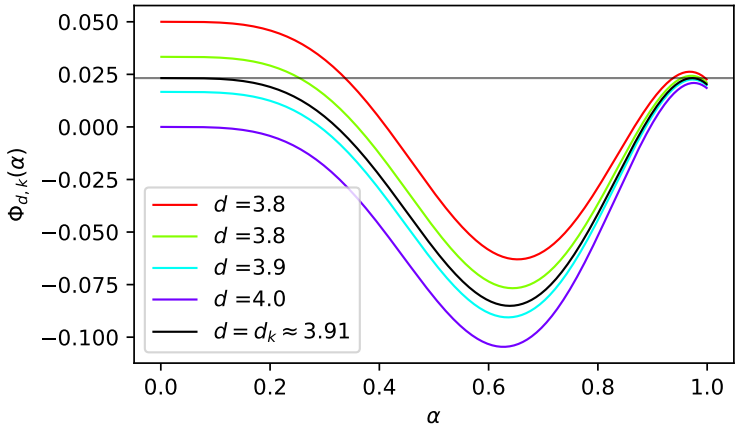


Figure 4.3: $\Phi_{d,4}$ from (4.2.1) for $k = 4$.

The following section concentrates of the two structural main ideas of the proof.

4.2.1 Pinning

One major problem in analysing $\boldsymbol{A}$ are the dependencies between columns of $\boldsymbol{A}$. More precisely, $\boldsymbol{A}$ contains so called "short linear relations" between variables. Intuitively, these relations pose as derived formulas. *Pinning* is the answer to this problem used in [34]. The idea of pinning goes back to Montanari [90] for Low Density Generator Matrix (LDGM) codes (among others) and has been successfully applied in other scenarios [36, 105].

We can transform $\boldsymbol{A}$ into an altered matrix $\boldsymbol{A}^\dagger$ by adding only a few rows with a single non-zero entry that *pins* exactly one variable to zero. This operation will be minimal enough for the matrix not to distort it to much, but suffices to "shatter" most of the short linear relations.

Following [59] we call a set of columns J a *relation of* A if there exists a linear combination of rows with J as the set of non zero entries. Further, a variable is called *frozen in* A, if $\{j\}$ is a relation of A. This characterizes exactly the columns that equal zero in every kernel vector. We write $\mathcal{F}(A)$ for the set of frozen variables in A and say that $J \neq \emptyset$ is a *proper relation* of A if $J \setminus \mathcal{F}(A)$ is a relation of A. Finally, we say that A is (δ, ℓ)*-free* if A possesses fewer than $\delta\binom{N}{h}$ proper relations I of size $|I| = h$ for any $2 \leq h \leq \ell$. For a k-XORSAT formula this means that one can only derive few short clauses that do not only contain variables that are already completely determined by the rest fo the formula.

For an integer $t \geq 0$ let $A[t]$ denotes the pinned matrix obtained from A by adding t new rows, each containing exactly one non-zero entry at a random position. The following lemma provides that the pinned matrix indeed only contains few short relations.

Lemma 4.2.1 ([59, Prop. 2.4], [34, Lemma 2.1])**.** *For any $\delta > 0, \ell > 0$ there exists $T_0 = O(\ell^3/\delta^4) > 0$ such that for any $T \geq T_0$ and any matrix A for a random $\boldsymbol{t} \in [T]$ we have $\mathbb{P}[A[\boldsymbol{t}]$ is (δ, ℓ)-free$] > 1 - \delta$.*

Thus, with $T = \lceil \ln n \rceil$, the matrix $\boldsymbol{A}^\dagger = \boldsymbol{A}[\boldsymbol{t}]$ is (ω^{-1}, ω)-free with $\omega = \lceil \ln \ln n \rceil$ w.h.p. In effect, the short ranged dependencies between variables will vanish, allowing us to characterize the set of frozen variables using Warning Propagation.

4.2.2 Warning Propagation

Warning Propagation update provides a heuristic fixed point equation for these messages. We distinguish between *standard* Warning Propagation messages and WP-*updates* and follow [28]. Former only depend on the true states of the variables in the matrix kernel. Thus, these message are not updated but formalize what the messages are *supposed* to encode. The latter are the update messages according to Section 2.4. We prove, the WP-updates in a fixed point essentially equal the standard messages. This provides that these WP-update messages are indeed a appropriate measure to characterize the kernel.

The WP-standard messages encode the true interactions regarding the kernel vectors. A variable node i 'tells' a factor node a that i is frozen (sends $\mathtt{f}$) if i is frozen and a is not the cause of it, i.e. if i is frozen even if a is removed. A factor node a sends $\mathtt{f}$ to i if a is the cause of i being frozen, i.e. if i is frozen even if any other adjacent factor node of i is removed [34, Eq. (2.1)].

$$\mathtt{m}_{v_j \to a_i}(A) = \begin{cases} \mathtt{f} & \text{if } j \in \mathcal{F}(A \setminus \{a_i\}) \\ \mathtt{u} & \text{otherwise} \end{cases} \tag{4.2.3}$$

$$\mathtt{m}_{a_i \to v_j}(A) = \begin{cases} \mathtt{f} & \text{if } v_j \in \mathcal{F}(A \setminus (\partial v_j \setminus \{a_i\})) \\ \mathtt{u} & \text{otherwise} \end{cases} \qquad (i \in [M],\, j \in [N]). \tag{4.2.4}$$

WP-updates Intuitively, a factor node a, that corresponds to an k-XORSAT clause, forces a variable v to zero if its' other neighbours are forced to zero. Otherwise, if at least one other neighbouring variable node is unfrozen, this variable node still can satisfy a for every value of v. Similarly, a variable node is frozen already if only one clause forces it to be frozen and unfrozen otherwise [34, Eq. (2.2)]. The messages are visualized in Fig. 4.4.

$$\hat{\mathtt{m}}_{v_j \to a_i} = \begin{cases} \mathtt{f} & \text{if } \exists a_h \in \partial v_j \setminus \{a_i\} : \mathtt{m}_{a_h \to v_j} = \mathtt{f}, \\ \mathtt{u} & \text{otherwise}, \end{cases} \tag{4.2.5}$$

$$\hat{\mathtt{m}}_{a_i \to v_j} = \begin{cases} \mathtt{f} & \text{if } \forall x_h \in \partial a_i \setminus \{v_j\} : \mathtt{m}_{x_h \to a_i} = \mathtt{f}, \\ \mathtt{u} & \text{otherwise}. \end{cases} \tag{4.2.6}$$

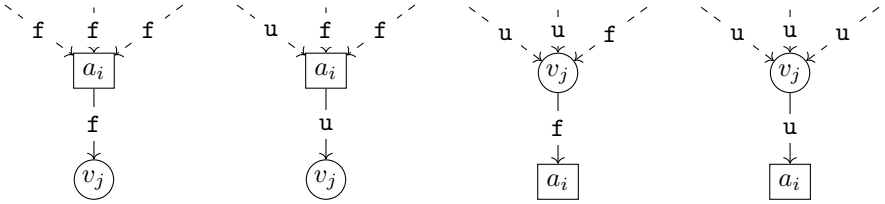


Figure 4.4: a visualisation of the Warning Propagation messages defined in (4.2.5)—(4.2.6)

The States are an additional marker based on the incoming messages to distinguish between frozen, unfrozen and *slush* variables that are barely frozen/unfrozen. This distinction categorizes the vertices regarding the occurring adjacent *combination* of messages at the edges [34, Eq. (2.3 – 2.4)].

$$\mathtt{m}_{v_j} = \begin{cases} \mathtt{f} & \text{if } \mathtt{m}_{a \to v_j} = \mathtt{f} \text{ for at least two } a \in \partial v_j, \\ \mathtt{s} & \text{if } \mathtt{m}_{a \to v_j} = \mathtt{f} \text{ for precisely one } a \in \partial v_j, \\ \mathtt{u} & \text{otherwise}, \end{cases} \tag{4.2.7}$$

$$\mathtt{m}_{a_i} = \begin{cases} \mathtt{f} & \text{if } \mathtt{m}_{v \to a_i} = \mathtt{f} \text{ for all } v \in \partial a_i, \\ \mathtt{s} & \text{if } \mathtt{m}_{v \to a_i} = \mathtt{f} \text{ for all but one } v \in \partial a_i, \\ \mathtt{u} & \text{otherwise}. \end{cases} \tag{4.2.8}$$

Coinciding messages The following proposition gives that the WP-updates, that encode only local interactions, coincide with the standard messages on a random matrix $\boldsymbol{A}^\dagger$ even though we are not limited to trees.

Proposition 4.2.2 ([34, Proposition 2.3]). *Let $d > 0, k \geq 3$. W.h.p. we have*

$$\sum_{i=1}^{m} \sum_{v_j \in \partial_{\boldsymbol{A}^\dagger} a_i} \mathbb{1}\left\{\mathfrak{m}_{v_j \to a_i}(\boldsymbol{A}^\dagger) \neq \hat{\mathfrak{m}}_{v_j \to a_i}(\boldsymbol{A}^\dagger)\right\} + \mathbb{1}\left\{\mathfrak{m}_{a_i \to v_j}(\boldsymbol{A}^\dagger) \neq \hat{\mathfrak{m}}_{a_i \to v_j}(\boldsymbol{A}^\dagger)\right\} = o(n), \tag{4.2.9}$$

$$\left|\left\{j \in [n] : \mathfrak{m}_{v_j}(\boldsymbol{A}^\dagger) \neq \mathtt{u}\right\} \triangle \mathcal{F}(\boldsymbol{A}^\dagger)\right| = o(n), \tag{4.2.10}$$

$$\sum_{s \in \mathbb{F}_q} \sum_{\ell \geq 0} \left| \sum_{j=1}^{n} \mathbb{1}\{d_{\boldsymbol{A}^\dagger}(v_j) = \ell,\, \mathfrak{m}_{v_j}(\boldsymbol{A}^\dagger) = \mathtt{u}\} \left(\mathbb{1}\{\boldsymbol{\sigma}_j^\dagger = s\} - 1/q\right) \right| = o(n). \tag{4.2.11}$$

This proposition allows us to characterize kernel vectors using local interactions.

4.2.3 Quenched Argument - Number of Messages

The goal of the quenched argument is to give a quantitative characterization of the kernel. More precisely, we characterize the number of pairs of messages that occur in the fixed point of WP in terms of the fraction of frozen variables in the fixed point. Let $\ell = (\ell_{\mathtt{uu}}, \ell_{\mathtt{uf}}, \ell_{\mathtt{fu}}, \ell_{\mathtt{ff}}) \in \mathbb{Z}^4_{\geq 0}$ be the vector representing the number of message combinations, where $\ell_{\mathtt{uf}}$ equals the number of edges with message combination `u` (incoming) `f` (outgoing), etc. Further define Δ_ℓ and Γ_ℓ as the number of variable resp. factor nodes that receive/send out messages according to ℓ.

We will show that Δ_ℓ and Γ_ℓ can be derived from a Galton Watson tree that mimics the Tanner graph. First, the following definitions of $\bar{\delta}$ and $\bar{\gamma}$ gives the distribution of states for factor and variable nodes in this Galton Watson tree.

Let's first consider the factor nodes state and assume, that each incoming message is `f` independently with probability α. Since a factor node has exactly k ingoing messages, the distribution over the states are as follows [34, Eq. (2.14)]:

$$\bar{\gamma}(\alpha, \mathtt{u}) = 1 - k(1-\alpha)\alpha^{k-1} - \alpha^k, \tag{4.2.12}$$

$$\bar{\gamma}(\alpha, \mathtt{s}) = k(1-\alpha)\alpha^{k-1}, \tag{4.2.13}$$

$$\bar{\gamma}(\alpha, \mathtt{f}) = \alpha^k. \tag{4.2.14}$$

Next we consider the variable nodes, while assuming that the ingoing variable messages of the second neighbourhood again are `f` independently with probability α. Further, assume that the degree of a variable node $\mathrm{Po}(d)$ distributed. Then we obtain the following distribution over states [34, Eq. (2.15)]:

$$\bar{\delta}(\alpha, \mathtt{u}) = \mathbb{P}[\mathrm{Po}(d\mathbb{P}[\mathrm{Bin}(k-1, \alpha)] = k-1) = 0] = \exp(-d\alpha^{k-1}), \tag{4.2.15}$$

$$\bar{\delta}(\alpha, \mathtt{s}) = d\alpha^{k-1}\exp(-d\alpha^{k-1}), \tag{4.2.16}$$

$$\bar{\delta}(\alpha, \mathtt{f}) = 1 - \exp(-d\alpha^{k-1})(1 + d\alpha^{k-1}),. \tag{4.2.17}$$

Now, we calculate for each variable and factor node the probability to both, have a specific given state and to satisfy an message vector ℓ. Each state gives us different restriction or ranges of freedom for the adjacent messages, that we need to consider. With $\mathrm{Po}_{\geq 2}(\lambda)$ and $\mathrm{Bin}_{\geq 2}(N, p)$ denoting the conditional Poisson/Binomial distributions given an outcome of at least two, we obtain the following expressions [34, Eq. (2.16 – 2.21)]:

$$\bar{\Delta}_{\mathtt{u},\ell}(\alpha) = \bar{\delta}(\alpha, \mathtt{u})\mathbb{1}\{\ell \in \mathcal{D}(\mathtt{u})\}\mathbb{P}\big[\mathrm{Po}(d(1-\alpha^{k-1})) = \ell_{\mathtt{uu}}\big], \tag{4.2.18}$$

$$\bar{\Delta}_{\mathtt{s},\ell}(\alpha) = \bar{\delta}(\alpha, \mathtt{s})\mathbb{1}\{\ell \in \mathcal{D}(\mathtt{s})\}\mathbb{P}\big[\mathrm{Po}(d(1-\alpha^{k-1})) = \ell_{\mathtt{uf}}\big], \tag{4.2.19}$$

$$\bar{\Delta}_{\mathtt{f},\ell}(\alpha) = \bar{\delta}(\alpha, \mathtt{f})\mathbb{1}\{\ell \in \mathcal{D}(\mathtt{f})\}\mathbb{P}\big[\mathrm{Po}_{\geq 2}(d\alpha^{k-1}) = \ell_{\mathtt{ff}}\big]\mathbb{P}\big[\mathrm{Po}(d(1-\alpha^{k-1})) = \ell_{\mathtt{uf}}\big], \tag{4.2.20}$$

$$\bar{\Gamma}_{\mathtt{u},\ell}(\alpha) = \bar{\gamma}(\alpha, \mathtt{u})\mathbb{1}\{\ell \in \mathcal{G}(\mathtt{u})\}\mathbb{P}[\mathrm{Bin}_{\geq 2}(k, 1-\alpha) = \ell_{\mathtt{uu}}], \tag{4.2.21}$$

$$\bar{\Gamma}_{\mathtt{s},\ell}(\alpha) = \bar{\gamma}(\alpha, \mathtt{s})\mathbb{1}\{\ell \in \mathcal{G}(\mathtt{s})\}, \tag{4.2.22}$$

$$\bar{\Gamma}_{\mathtt{f},\ell}(\alpha) = \bar{\gamma}(\alpha, \mathtt{f})\mathbb{1}\{\ell \in \mathcal{G}(\mathtt{f})\}. \tag{4.2.23}$$

The following Proposition ensures that above calculation based on some intuitive assumptions indeed coincides with the true values.

Proposition 4.2.3 ([34, Proposition 2.4]). *Let $d > 0, k \geq 3$. Then w.h.p. for all but $o(n)$ adjacent pairs v_j, v_i the fixed point equations* (4.2.5),(4.2.6) *hold. Moreover for all ℓ*

$$\mathbb{E}\left||\Delta_\ell| - n\left|\bar{\Delta}_\ell(\boldsymbol{\alpha})\right|\right| + \mathbb{E}\left||\Gamma_\ell| - m\bar{\Gamma}_\ell(\boldsymbol{\alpha})\right| = o(n).$$

Finally, for all but $o(n)$ exceptions variable v_j is frozen iff $\mathtt{m}_{a_i \to v_j} = \mathtt{f}$ for some $a_i \in \partial v_j$.

The proof of Proposition 4.2.3 is based on coupling arguments and does not reveal the likely value of $\boldsymbol{\alpha}$.

4.2.4 Annealed Analysis - Moment Calculation

The quenched argument gives us quantitative information about the kernel given in terms of the yet unknown random variable $\boldsymbol{\alpha}$. In the last step we prove that $\boldsymbol{\alpha} = o(1)$ w.h.p. if $d < (1-\varepsilon)d_k/k$ using our knowledge about the typical shape of kernel vectors. To this end we prove that any WP fixed point with $\Omega(n)$ frozen variables leads to to few kernel vectors. First, we estimate the expected number of WP fixed points with an α-fraction of frozen variables. This turns out to be sub-exponential for any $0 \leq \alpha \leq 1$. Next, we estimate the number of kernel vectors $\boldsymbol{X}_{\boldsymbol{\alpha}}$ that extend a specific α-WP fixed point. Combining this with Proposition 4.2.3 gives us w.h.p. $\left|\ker \boldsymbol{A}^\dagger\right| \sim \boldsymbol{X}_{\boldsymbol{\alpha}}$. Further, let $\mathfrak{D}$ be the σ-algebra given by the degree-sequence of the factor graph. The following proposition gives a upper bound on the expectation of $\boldsymbol{X}_\alpha$ for any $0 \leq \alpha \leq 1$ in terms the Bethe free entropy $\Phi_{d,k}$ from (4.2.1).

Proposition 4.2.4 ([34, Proposition 2.5]). *Let $d > 0, k \geq 3$. W.h.p. for all $\alpha \in [0,1]$ we have*

$$\mathbb{E}\left[\boldsymbol{X}_\alpha \mid \mathfrak{D}\right] \leq q^{n\Phi_{d,k}(\alpha)+o(n)}.$$

Chapter 5

Patient Zero

Spreading processes in graphs are used to model a multitude of real world settings. In epidemiology these models are used to study the spreading of an epidemic due to close interactions modelled by an underlying graph [19, 79]. Similarly, the propagation of viruses in local computer networks (i.e. in a software company) are modelled like this. Another natural application is to study the spreading of (miss-) information in (social) networks [13, 78, 88, 98, 110].

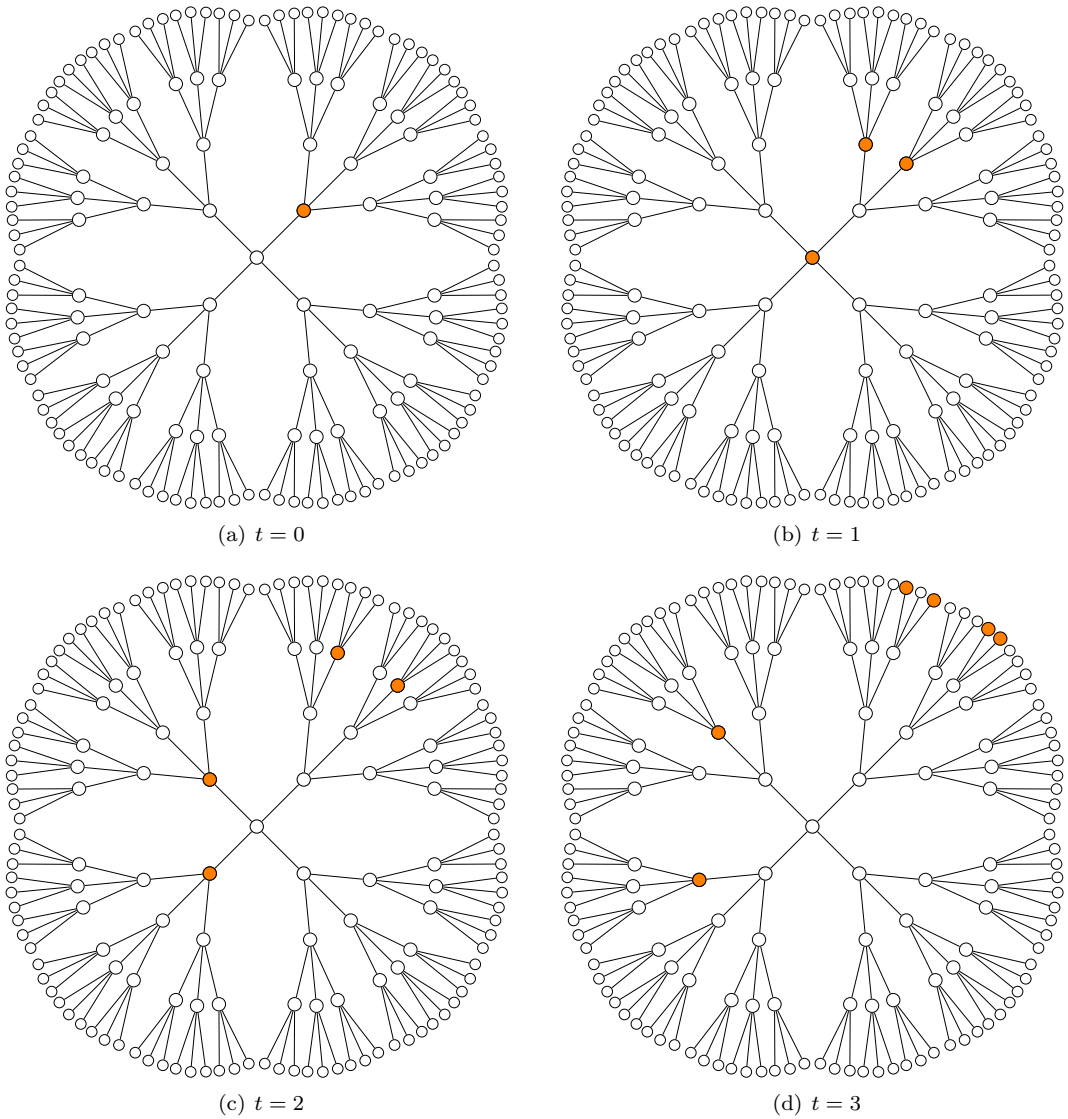
The general model works intuitively works as follows. Given a graph, in the beginning only a very small subset of vertices is "infected", coinciding with the very few individuals with a new information, a rumour, or infected with a relatively new disease. Now at each time, an infected individual can infect their neighbours, thus spreading the rumour/ the disease through the underlying network.

For these propagation models, *forward processes* have been studied intensively in the past, including research on the number of infected or influence on the vertices in a social network such that those processes are now quite well understood in simple networks [20, 97]. However, we are interested in the *backwards problem*, where the goal is to infer the *beginning* of an epidemic commonly referred to as the *patient zero problem*. This inference problem was studied rigorously on the SI-model from epidemiology [115]. In this model, once a node is infected, it stays infected and can infect randomly chosen neighbours.

5.1 Model

The contained result studies a specialized case of the discrete SIR model [77]. Given a graph $G = (V, E)$, each vertex has at each time one of three states: susceptible (S), infected (I) or recovered (R). At the beginning of the process, each vertex is susceptible with the exception of the patient zero $\boldsymbol{\omega}$ that is infected. Then, at the beginning of each step every infected vertex first infects each of its neighbours with a rate p_i and recovers with probability p_r. Once a vertex is recovered, it will never get infected again.

The *independent cascade model* is a special case of the SIR model where $p_r = 1$ [78]. Hence, every infected node has only one time step to infect each neighbours. Furthermore, in the studied setting the observer only receives the set of currently infected individuals $\boldsymbol{X}^\star$ at the time t but does **not** know the time of observation t and neither the already recovered nodes. Now, the task of the observer is to give the most likely origin of the infection process.



(a) $t = 0$ (b) $t = 1$

(c) $t = 2$ (d) $t = 3$

Figure 5.1: Example process of the ICM, the orange vertices mark the currently infected individuals.

5.2 Prior Work

Forward propagation processes in graph networks like the epidemic models [19, 77, 79, 128], rumour spreading [61, 88, 98, 110] and information cascades [78, 130] have been studied extensively in the past. However, rigorous results from the corresponding inference problem, i.e. identifying the source of a propagation process, are rather scarce. For the SI model rigorous contributions include [76, 114, 115]. [114, 115] prove that on some infinite acyclic networks like super-critical Galton-Watson processes, d-regular trees, and geometric graphs, approximate inference of the source is possible given certain expansion properties. Further, the estimate of the rumours source is unlikely to be far from the true source.

Moreover, there has been studies concerning the more general SIR model [131, 132] which contains the ICM as one special case. On this model, [131] study a etsimator based on the Jordan centrality for d-regular trees. On the heuristic side, there has been extensive studies on this problem in multiple settings based on simulations [9, 18, 26, 71, 73, 104, 124].

5.3 Contribution

The results of this section from the paper

Inference of a Rumor's Source in the Independent Cascade Model

by Petra Berenbrink, Max Hahn-Klimroth, Dominik Kaaser, Lena Krieg and Malin Rau published in *Proceedings of the 39th Conference on Uncertainty in Artificial Intelligence* provides two main results for d-regular and Galton-Watson trees. For both models we first pin down the threshold for the parameters where strong and weak detection is possible/ is not possible. Second, we provide an efficient algorithm that returns the the MAP solution and provide that this algorithms solves both weak and strong detection.

For all results we rely on the following fact that guarantees that the MAP solution of the posteriori information is the

Fact 5.3.1 ([14, Theorem 2.1])**.** *Let $G = (V, E)$ be an arbitrary network and fix an arbitrary step t. Let $\boldsymbol{X}^\star$ be the set active vertices in step t. For any $X \subseteq V$*

$$\arg\max_{v \in V} \mathbb{P}\left(\boldsymbol{X}^\star = X \mid \boldsymbol{\omega} = v\right) = \arg\max_{v \in V} \mathbb{P}\left(\boldsymbol{\omega} = v \mid \boldsymbol{X}^\star = X\right).$$

Let the set of candidates $\mathcal{C} = \{v \in V : \forall u, w \in I : d(u, v) = d(w, v)\}$ be the set of equidistant vertices to every vertex in I. Further, let $\boldsymbol{\omega}_c$ be the individual in $\mathcal{C}$ that is closest to the infected individuals. The following theorem shows that the threshold for detection in d-regular trees.

Theorem 11 (d-regular trees [14, Theorem 2.2])**.** *Let $G = (V, E)$ be an infinite d-regular tree and let $\boldsymbol{X}^\star$ be the set of active nodes generated by the ICM with spreading parameter p after $t = \omega(1)$ steps. Then, the following phase-transitions occur.*

- *If $(d-1) \cdot p \leq 1$, any estimator fails at weak detection with probability $1 - o_t(1)$. (We denote by $o_t(1)$ a quantity that tends to zero with $t \to \infty$.)*
- *If $1 < (d-1) \cdot p = \Theta(1)$ then the closest candidate $\boldsymbol{\omega}_c$ is the source of the rumour $\boldsymbol{\omega}$ with constant probability (weak detection). Furthermore, the probability that $\operatorname{dist}(\boldsymbol{\omega}_c, \boldsymbol{\omega}) > k$ is at most $\exp\left(-\Omega(k)\right)$.*
- *If $(d-1) \cdot p = \omega(1)$ then closest candidate $\boldsymbol{\omega}_c$ is the source of the rumour $\boldsymbol{\omega}$ with probability $1 - o_d(1)$ (strong detection).*

Further, the next theorem gives essentially the same result for Galton-Watson processes

Theorem 12 (Galton-Watson processes [14, Theorem 2.3])**.** *Let $G = (V, E)$ be an infinite tree generated by a $\mathrm{Po}(\lambda)$-Galton-Watson process. Let $\boldsymbol{X}^\star$ be the set of active nodes generated by the ICM with spreading parameter p after $t = \omega(1)$ steps. Then, the following phase-transition occurs.*

- *If $\lambda p \leq 1$, any estimator fails at weak detection with probability $1 - o_t(1)$.*
- *If $1 < \lambda p = \Theta(1)$, then the closest candidate $\boldsymbol{\omega}_c$ is the source of the rumour $\boldsymbol{\omega}$ with positive probability (weak detection). Furthermore, the probability that $\operatorname{dist}(\boldsymbol{\omega}_c, \boldsymbol{\omega}) > k$ is at most $\exp\left(-\Omega(k)\right)$.*
- *If $\lambda p = \omega(1)$, then closest candidate $\boldsymbol{\omega}_c$ is the source of the rumour $\boldsymbol{\omega}$ with probability $1 - o_\lambda(1)$ (strong detection).*

5.4 Proof Outline

The special case of the ICM model on trees allows one important structural observation. Given the set of individuals I, we know that $\boldsymbol{\omega}$ is equidistant to all individuals in I due to the one chance property given by $p_r = 1$ and hence part of $\mathcal{C}$. Further, we view at the infection process as a tree rooted in $\boldsymbol{\omega}$. If we observe at least one infected individual in a subtree of $\boldsymbol{\omega}$, we call this subtree *alive* at time t, else the infection has died out in this subtree. Both theorems basically characterize the following regimes:

- The process is alive in all subtrees of $\boldsymbol{\omega}$. Hence, there is only one possible candidate, $|\mathcal{C}| = 1$, then we can indeed recover the patient zero exactly, since there is only one possible candidate.

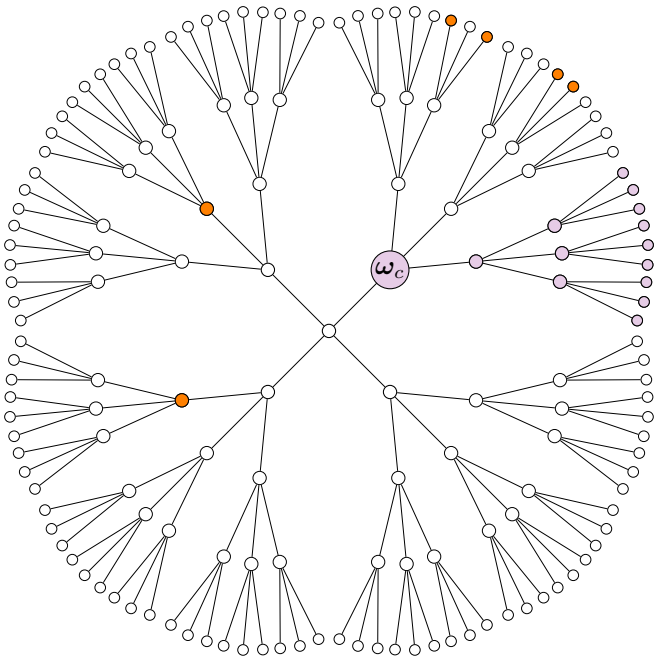


Figure 5.2: Example of Spreading. The labeled vertex $\boldsymbol{\omega}$ is the true origin of the rumour, the orange vertices are a possible observation at time $t = 3$, the set of violet vertices represent the set of possible candidates of the origin given the observation [14, Figure 1].

- The process has died out in at least one subtree of $\boldsymbol{\omega}$ but not in all subtrees. The set of candidates is one subtree of the true patient zero. Here, the most likely patient zero is the vertex that minimizes the distance to the set of infected nodes I and we can prove a exponential probability decay for the vertices further distant.
- The process died out in all subtrees of $\boldsymbol{\omega}$ and we observed nothing. Here, any vertex is equally likely to be the source of the rumour and we will fail w.h.p.

Hence, the proof comes down to analyse the underlying tree after being thinned out through the infected process and determine which parameters lead to which outcome of the possibilities above. Section 5.4 visualises a possible observation. Here, the orange node indicate the infected individuals we observe, whereas the violet vertices are all vertices that could in principle be the source. The (proposed) estimator would return MAP-estimate $\boldsymbol{\omega}_c$.

Chapter 6

Conclusion

This work contains sharp thresholds for multiple settings, as well as efficient algorithms that are optimal, i.e. match these thresholds. However, this work can, of course, be continued in every single field.

6.1 Group Testing

Even though sublinear noisy group testing is almost understood completely, an exact lower bound for exact recovery is still missing. Prior results suggest that the constant column result is information theoretically optimal. Further, for approximate recovery for sublinear noisy group testing, the success probability of Theorem 7 proves, that no test design and algorithm will succeed under this threshold of necessary tests w.h.p. However, we conjecture that actually no test design and algorithm is able to success with *any* positive probability.

For linear group testing, there are two adjacent open questions. First, for noisy linear group testing we solved the problem of exact recovery completely. Algorithmically, we always create an approximate solution in the first step. However, if one is interested in approximate recovery, the question of a sharp threshold to achieve this remains open. Second, for noiseless linear group testing, the general threshold needed for adaptive group testing is still largely unknown. We already know, that after α exceeds $\frac{1}{2}(3-\sqrt{5})$ individual testing is the best one can do [120]. Also, there exists a couple of algorithmic results for the whole regime. However, the question of a sharp threshold for the whole range of α remains open.

6.2 Sparse Random Matrices

The techniques used in the re-proof of the k-XORSAT threshold can be applied to other models as well. For example, one could try to use a similar (but adjusted) technique on generalized degree distributions. However, heuristic suggests that, for example, the Bethe free entropy on these generalized degree distribution does not behave as simple and predictable as in our case. In contrast to random k-XORSAT, where we only observe two maxima, we might as well observe arbitrary many maxima for generalised degree distributions.

6.3 Patient Zero

Our work on the patient zero problem has only covered a very specific case on two quite specific classes of graphs. Obviously, there is still a lot of work left to do. The general SIR model is still to be understood completely, even on restricted graph classes like trees. Moreover, even the ICM is not yet fully understood on more general graph classes. Both of these would be the first obvious generalisation one could work on.

Bibliography

[1] E. Abbe. Community detection and stochastic block models: recent developments. *Journal of Machine Learning Research*, 18(177):1–86, 2018.

[2] E. Abbe and C. Sandon. Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In *Proc. of 56th Symposium on Foundations of Computer Science FOCS*, pages 670–688. IEEE, 2015.

[3] E. Abbe and C. Sandon. Proof of the achievability conjectures for the general stochastic block model. *Communications on Pure and Applied Mathematics*, 71(7):1334–1406, 2018.

[4] D. Achlioptas and A. Coja-Oghlan. Algorithmic barriers from phase transitions. In *Proc. of 49th IEEE Symposium on Foundations of Computer Science, FOCS*, pages 793–802. IEEE Computer Society, 2008.

[5] M. Aldridge. Individual testing is optimal for nonadaptive group testing in the linear regime. *IEEE Transactions on Information Theory*, 65(4):2058–2061, 2019.

[6] M. Aldridge. Rates of adaptive group testing in the linear regime. *IEEE International Symposium on Information Theory (ISIT)*, pages 236–240, 2019.

[7] M. Aldridge, L. Baldassini, and O. Johnson. Group testing algorithms: Bounds and simulations. *IEEE Transactions on Information Theory*, 60(6):3671–3687, 2014.

[8] M. Aldridge, O. Johnson, and J. Scarlett. Group testing: An information theory perspective. *Foundations and Trends in Communications and Information Theory*, 15(3-4):196–392, 2019.

[9] M. Amoruso, D. Anello, V. Auletta, R. Cerulli, D. Ferraioli, and A. Raiconi. Contrasting the spread of misinformation in online social networks. *Journal of Artificial Intelligence Research*, 69:847–879, 2020.

[10] S. Arora and B. Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.

[11] P. Ayre, A. Coja-Oghlan, P. Gao, and N. Müller. The satisfiability threshold for random linear equations. *Combinatorica*, 40(2):179–235, 2020.

[12] J. Barbier, W.-K. Chen, D. Panchenko, and M. Sáenz. Performance of bayesian linear regression in a model with mismatch. *arXiv preprint arXiv:2107.06936*, 2021.

[13] R. Becker, F. Corò, G. D'Angelo, and H. Gilbert. Balancing spreads of influence in a social network. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, pages 3–10. AAAI Press, 2020.

[14] P. Berenbrink, M. Hahn-Klimroth, D. Kaaser, L. Krieg, and M. Rau. Inference of a rumor's source in the independent cascade model. In *Uncertainty in Artificial Intelligence, UAI*, volume 216 of *Proc. of Machine Learning Research*, pages 152–162. PMLR, 2023.

[15] T. Berger and V. I. Levenshtein. Asymptotic efficiency of two-stage disjunctive testing. *IEEE Transactions on Information Theory*, 48(7):1741–1749, 2002.

[16] E. Berlekamp, R. McEliece, and H. Van Tilborg. On the inherent intractability of certain coding problems (corresp.). *IEEE Transactions on Information theory*, 24(3):384–386, 2003.

[17] H. A. Bethe. Statistical theory of superlattices. *Proc. of the Royal Society of London. Series A-Mathematical and Physical Sciences*, 150(871):552–575, 1935.

[18] J. Bindi, A. Braunstein, and L. Dall'Asta. Predicting epidemic evolution on contact networks from partial observations. *PloS one*, 12:1–28, 2017.

[19] F. Brauer. Mathematical epidemiology: Past, present, and future. *Infectious Disease Modelling*, 2(2):113–127, 2017.

[20] T. Britton, S. Janson, and A. Martin-Löf. Graphs with specified degree distributions, simple epidemics, and local vaccination strategies. *Advances in Applied Probability*, 39(4):922–948, 2007.

[21] N. H. Bshouty. Optimal algorithms for the coin weighing problem with a spring scale. In *The 22nd Conference on Learning Theory COLT*, 2009.

[22] F. Camilli, P. Contucci, and E. Mingione. An inference problem in a mismatched setting: a spin-glass model with mattis interaction. *SciPost Physics*, 12(4):125, 2022.

[23] C. L. Chan, P. H. Che, S. Jaggi, and V. Saligrama. Non-adaptive probabilistic group testing with noisy measurements: Near-optimal bounds with efficient algorithms. In *Proc. of 49th Allerton Conference on Communication, Control, and Computing Allerton*, pages 1832–1839. IEEE, 2011.

[24] H.-B. Chen and F. K. Hwang. A survey on nonadaptive group testing algorithms through the angle of decoding. *Journal of Combinatorial Optimization*, 15:49–59, 2008.

[25] J. Chen and J. Scarlett. Exact thresholds for noisy non-adaptive group testing. pages 4644–4706, 2025.

[26] W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In *Proc. of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 199–208. Association for Computing Machinery, 2009.

[27] M. Cheraghchi and V. Nakos. Combinatorial group testing and sparse recovery schemes with near-optimal decoding time. *proc. 61st IEEE Symposium on Foundations of Computer Science, FOCS*, pages 1203–1213, 2020.

[28] A. Coja-Oghlan, O. Cooley, M. Kang, J. Lee, and J. B. Ravelomanana. The sparse parity matrix. *Advances in Combinatorics*, pages Paper No. 5, 68, 2023.

[29] A. Coja-Oghlan and C. Efthymiou. On independent sets in random graphs. *Random Structures & Algorithms*, 47(3):436–486, 2015.

[30] A. Coja-Oghlan, O. Gebhard, M. Hahn-Klimroth, and P. Loick. Information-theoretic and algorithmic thresholds for group testing. *IEEE Transactions on Information Theory*, 66(12):7911–7928, 2020.

[31] A. Coja-Oghlan, O. Gebhard, M. Hahn-Klimroth, and P. Loick. Optimal group testing. *Combinatorics, Probability and Computing*, 30(6):811–848, 2021.

[32] A. Coja-Oghlan, O. Gebhard, M. Hahn-Klimroth, A. S. Wein, and I. Zadik. Statistical and computational phase transitions in group testing. In *Conference on Learning Theory*, pages 4764–4781. PMLR, 2022.

[33] A. Coja-Oghlan, M. Hahn-Klimroth, L. Hintze, D. Kaaser, L. Krieg, M. Rolvien, and O. Scheftelowitsch. Noisy group testing via spatial coupling. *Combinatorics, Probability and Computing*, 34(2):210–258, 2025.

[34] A. Coja-Oghlan, M. Kang, L. Krieg, and M. Rolvien. The k-XORSAT threshold revisited. *Electronic Journal of Combinatorics*, 31(2), 2024.

[35] A. Coja-Oghlan, L. Krieg, J. C. Lawnik, and O. Scheftelowitsch. Bad local minima exist in the stochastic block model. *Journal of Statistical Physics*, 191(11):148, 2024.

[36] A. Coja-Oghlan, F. Krzakala, W. Perkins, and L. Zdeborová. Information-theoretic thresholds from the cavity method. *Advances in Mathematics*, 333:694–795, 2018.

[37] S. Cook. The complexity of theorem-proving procedures. In *Proc. of the 3rd ACM Symposium on Theory of computing STOC*, pages 151–158, 1971.

[38] G. F. Cooper. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial Intelligence*, 42(2):393–405, 1990.

[39] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84:066106, 2011.

[40] M. Dietzfelbinger, A. Goerdt, M. Mitzenmacher, A. Montanari, R. Pagh, and M. Rink. Tight thresholds for cuckoo hashing via XORSAT. In *Proc. of 37th International Colloquium ICALP*, pages 213–225. Springer, 2010.

[41] J. Ding, A. Sly, and N. Sun. Proof of the satisfiability conjecture for large k. *Annals of Mathematics. Second Series.*, 196(1):1–388, 2022.

[42] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[43] D. L. Donoho, A. Javanmard, and A. Montanari. Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing. *IEEE Transactions on Information Theory*, 59(11):7434–7464, 2013.

[44] R. Dorfman. The Detection of Defective Members of Large Populations. *The Annals of Mathematical Statistics*, 14(4):436 – 440, 1943.

[45] D.-Z. Du and F. K. Hwang. *Combinatorial group testing and its applications*, volume 12. World Scientific, 1999.

[46] O. Dubois and J. Mandler. The 3-XORSAT Threshold. In *Proc. of the 43rd Symposium on Foundations of Computer Science*, page 769–778, 2002.

[47] A. El Alaoui, A. Ramdas, F. Krzakala, L. Zdeborová, and M. I. Jordan. Decoding from pooled data: Phase transitions of message passing. *IEEE Transactions on Information Theory*, 65(1):572–585, 2018.

[48] J. C. Emmanuel, M. T. Bassett, H. J. Smith, and J. Jacobs. Pooling of sera for human immunodeficiency virus (HIV) testing: an economical method for use in developing countries. *Journal of clinical pathology*, 41(5):582–585, 1988.

[49] M. Etscheid and H. Röglin. Smoothed analysis of local search for the maximum-cut problem. *ACM Transactions on Algorithms*, 13(2):25:1–25:12, 2017.

[50] U. Feige and A. Lellouche. Quantitative group testing and the rank of random matrices. *arXiv preprint arXiv:2006.09074*, 2020.

[51] E. Friedgut and J. Bourgain. Sharp thresholds of graph properties, and the k-SAT problem. *Journal of the AMS*, 12(4):1017–1054, 1999.

[52] A. Frieze and M. Karoński. *Introduction to random graphs*. Cambridge University Press, 2015.

[53] A. M. Frieze. On the independence number of random graphs. *Discrete Mathematics*, 81(2):171–175, 1990.

[54] R. Gallager. Low-density parity-check codes. *IRE Transactions on Information Theory*, 8(1):21–28, 1962.

[55] D. Gamarnik, C. Moore, and L. Zdeborová. Disordered systems insights on computational hardness. *Journal of Statistical Mechanics: Theory and Experiment*, 2022, 2022.

[56] D. Gamarnik and M. Sudan. Limits of local algorithms over sparse random graphs. In *Proc. of Innovations in Theoretical Computer Science ITCS*, pages 369–376. ACM, 2014.

[57] D. Gamarnik and M. Sudan. Performance of sequential local algorithms for the random NAE-k-SAT problem. *SIAM Journal on Computing*, 46(2):590–619, 2017.

[58] E. Gardner and B. Derrida. Three unfinished works on the optimal storage capacity of networks. *Journal of Physics A: Mathematical and General*, 22(12):1983, 1989.

[59] O. Gebhard, O. Johnson, P. Loick, and M. Rolvien. Improved bounds for noisy group testing with constant tests per item. *IEEE Transactions on Information Theory*, 68(4):2604–2621, 2021.

[60] A. Giurgiu, N. Macris, and R. Urbanke. Spatial coupling as a proof technique and three applications. *IEEE Transactions on Information Theory*, 62(10):5281–5295, 2016.

[61] W. Goffman and V. Newill. Generalization of epidemic theory: An application to the transmission of ideas. *Nature*, 204(4955):225–228, 1964.

[62] V. Guruswami, A. Rudra, and M. Sudan. Essential coding theory. *Draft available at https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/*, 2(1), 2012.

[63] K. Haar, T. Meyer, S. Desai, M. Thamm, M. a. d. Heiden, V. Bremer, and O. Hamouda. Low sensitivity of pooled chlamydia testing in a sample of the young german general population. *Journal of US-China Medical Science*, 8(10), 2011.

[64] R. W. Hamming. Error detecting and error correcting codes. *The Bell System Technical Journal*, 29(2):147–160, 1950.

[65] L. Hintze, L. Krieg, O. Scheftelowitsch, and H. Zhu. Noisy linear group testing: Exact thresholds and efficient algorithms. *arXiv preprint arXiv:2411.03839*, 2024. to appear in Proc. of 38th Annual Conference on Learning Theory (COLT 2025).

[66] P. W. Holland, K. B. Laskey, and S. Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

[67] F. d. Hollander and O. Jovanovski. Glauber dynamics on the Erdős Rényi random graph. In *In and Out of Equilibrium 3: Celebrating Vladas Sidoravicius*, pages 519–589. Springer, 2021.

[68] E. S. Hong, R. E. Ladner, and E. A. Riskin. Group testing for wavelet packet image compression. *Proc. IEEE Data Compression Conference DCC*, pages 73–82, 2001.

[69] S. B. Hopkins and D. Steurer. Efficient bayesian estimation from few samples: community detection and related problems. In *Proc. of 58th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 379–390. IEEE, 2017.

[70] F. K. Hwang. A method for detecting all defective members in a population by group testing. *Journal of the American Statistical Association*, 67(339):605–608, 1972.

[71] A. Jain, V. S. Borkar, and D. Garg. Fast rumor source identification via random walks. *Social network analysis and mining*, 6(1):62:1–62:13, 2016.

[72] F. K. Jasima, N. Tutor, and P. Subhashini. Comparative evaluation of elisa & clia screening assays in the effective detection of hiv infection in blood donor samples. *International Journal of Health Sciences*, 2022.

[73] F. Ji and W. P. Tay. An algorithmic framework for estimating rumor sources with different start times. *IEEE Transactions on Signal Processing*, 65(10):2517–2530, 2017.

[74] O. Johnson, M. Aldridge, and J. Scarlett. Performance of group testing algorithms with near-constant tests per item. *IEEE Transactions on Information Theory*, 65(2):707–723, 2018.

[75] D. Kaplan. *Bayesian statistics for the social sciences*. Guilford Publications, 2023.

[76] S. J. Kazemitabar and A. A. Amini. Approximate identification of the optimal epidemic source in complex networks. In *Proc. of NetSci-X 2020: Sixth International Winter School and Conference on Network Science*, pages 107–125. Springer, 2020.

[77] J. M. Keeling and P. Rohani. *Modeling Infectious Diseases in Humans and Animals*. Princeton University Press, 2008.

[78] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proc. of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.

[79] W. O. Kermack and A. G. McKendrick. A contribution to the mathematical theory of epidemics. *Proc. of the Royal Society of London. Series A*, 115(772):700–721, 1927.

[80] H. Kesten and B. Stigum. Limit theorems for decomposable multi-dimensional galton-watson processes. *Journal of Mathematical Analysis and Applications*, 17(2):309–338, 1967.

[81] F. Krzakala, M. Mézard, F. Sausset, Y. Sun, and L. Zdeborová. Statistical-physics-based reconstruction in compressed sensing. *Physical Review X*, 2(2):021005, 2012.

[82] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, 2001.

[83] S. Kudekar and H. D. Pfister. The effect of spatial coupling on compressive sensing. In *Proc. of 48th Allerton Conference on Communication, Control, and Computing Allerton*, pages 347–353. IEEE, 2010.

[84] S. Kudekar, T. Richardson, and R. L. Urbanke. Spatially coupled ensembles universally achieve capacity under belief propagation. *IEEE Transactions on Information Theory*, 59(12):7761–7813, 2013.

[85] S. Kudekar, T. J. Richardson, and R. L. Urbanke. Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC. *IEEE Transactions on Information Theory*, 57(2):803–834, 2011.

[86] S. Kumar, A. J. Young, N. Macris, and H. D. Pfister. Threshold saturation for spatially coupled LDPC and LDGM codes on BMS channels. *IEEE Transactions on Information Theory*, 60(12):7389–7415, 2014.

[87] E. P. K. Kılbaş, I. Kilbas, and I. H. Ciftci. A meta-analysis on the comparison of the sensitivity of three test methods used in the diagnosis of COVID-19. *Journal of Kermanshah University of Medical Sciences*, 2022.

[88] K. Lerman and R. Ghosh. Information contagion: An empirical study of the spread of news on digg and twitter social networks. In *Proc. of the Fourth International Conference on Weblogs and Social Media, ICWSM*. The AAAI Press, 2010.

[89] J. H. Liang, V. Ganesh, K. Czarnecki, and V. Raman. SAT-based analysis of large real-world feature models is easy. *Proc. of the 19th International Conference on Software Product Line*, 2015.

[90] A. Montanari. Estimating random variables from random sparse observations. *European Transactions on Telecommunications*, 19(4):385–403, 2008.

[91] C. Moore. The computer science and physics of community detection: Landscapes, phase transitions, and hardness. *Bull. EATCS*, 121, 2017.

[92] E. Mossel, J. Neeman, and A. Sly. A proof of the block model threshold conjecture. *Combinatorica*, 38(3):665–708, 2018.

[93] L. Mutesa, P. Ndishimye, Y. Butera, J. Souopgui, A. Uwineza, R. Rutayisire, E. L. Ndoricimpaye, E. Musoni, N. Rujeni, T. Nyatanyi, et al. A pooled testing strategy for identifying SARS-CoV-2 at low prevalence. *Nature*, 589(7841):276–280, 2021.

[94] M. Mézard and A. Montanari. *Information, Physics, and Computation*. Oxford University Press, 2009.

[95] D. Nam, A. Sly, and Y. Sohn. One-step replica symmetry breaking of random regular NAE-SAT II. *Communications in Mathematical Physics*, 405(3):61, 2024.

[96] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM journal on computing*, 24(2):227–234, 1995.

[97] P. Neal. SIR epidemics on a bernoulli random graph. *Journal of Applied Probability*, 40(3):779–782, 2003.

[98] M. Nekovee, Y. Moreno, G. Bianconi, and M. Marsili. Theory of rumour spreading in complex social networks. *Physica A: Statistical Mechanics and its Applications*, 374(1):457—470, 2007.

[99] J. Pearl. *Reverend Bayes on inference engines: A distributed hierarchical approach*. Cognitive Systems Laboratory, School of Engineering and Applied Science, 1982.

[100] R. Peierls. Statistical theory of superlattices with unequal concentrations of the components. *Proc. of the Royal Society of London. Series A-Mathematical and Physical Sciences*, 154(881):207–222, 1936.

[101] B. Pittel and G. Sorkin. The satisfiability threshold for k-XORSAT. *Combinatorics, Probability and Computing*, 25(2):236–268, 2016.

[102] F. Pourkamali and N. Macris. Mismatched estimation of rank-one symmetric matrices under gaussian noise. *arXiv preprint arXiv:2107.08927*, 2021.

[103] F. Pourkamali and N. Macris. Mismatched estimation of non-symmetric rank-one matrices under gaussian noise. In *Proc. of IEEE International Symposium on Information Theory, ISIT*, pages 1288–1293. IEEE, 2022.

[104] B. A. Prakash, J. Vreeken, and C. Faloutsos. Efficiently spotting the starting points of an epidemic in a large graph balance. *Knowledge and Information Systems*, 38(1):35–59, 2014.

[105] P. Raghavendra and N. Tan. Approximating CSPs with global cardinality constraints using SDP hierarchies. In *Proc. of the 23rd ACM-SIAM Symposium on Discrete Algorithms SODA*, pages 373–387. SIAM, 2012.

[106] M. Rahman and B. Virág. Local algorithms for independent sets are half-optimal. *The Annals of Probability*, 45(2):1543–1577, 2017.

[107] I. S. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8(2):300–304, 1960.

[108] E. Richard and A. Montanari. A statistical model for tensor PCA. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems*, pages 2897–2905, 2014.

[109] T. Richardson and R. Urbanke. *Modern coding theory*. Cambridge university press, 2008.

[110] A. Sadilek, H. A. Kautz, and V. Silenzio. Modeling spread of disease from social interactions. In *Proc. of the Sixth International Conference on Weblogs and Social Media*. The AAAI Press, 2012.

[111] J. Scarlett. Noisy adaptive group testing: Bounds and algorithms. *IEEE Transactions on Information Theory*, 65(6):3646–3661, 2019.

[112] J. Scarlett and V. Cevher. Phase transitions in group testing. *Proc. 27th ACM-SIAM Symposium on Discrete algorithms*, pages 40–53, 2016.

[113] J. Scarlett and O. Johnson. Noisy non-adaptive group testing: A (near-) definite defectives approach. *IEEE Transactions on Information Theory*, 66(6):3775–3797, 2020.

[114] D. Shah and T. Zaman. Detecting sources of computer viruses in networks: theory and experiment. In *Proc. of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pages 203–214. ACM, 2010.

[115] D. Shah and T. Zaman. Rumor centrality: a universal source detector. In *Proc. of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pages 199–210. ACM, 2012.

[116] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:623–656, 1948.

[117] D. A. Spielman and S.-H. Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *Journal of the ACM (JACM)*, 51(3):385–463, 2004.

[118] R. E. Tarjan. Amortized computational complexity. *SIAM Journal on Algebraic Discrete Methods*, 6(2):306–318, 1985.

[119] T. Toptan, L. Eckermann, A. E. Pfeiffer, S. Hoehl, S. Ciesek, C. Drosten, and V. M. Corman. Evaluation of a SARS-CoV-2 rapid antigen test: Potential to help reduce community spread? *Journal of Clinical Virology*, 135:104713, 2021.

[120] P. Ungar. The cutoff point for group testing. *Communications on Pure and Applied Mathematics*, 13:49–54, 1960.

[121] L. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189–201, 1979.

[122] S. Verdú. Mismatched estimation and relative entropy. *IEEE Transactions on Information Theory*, 56(8):3712–3720, 2010.

[123] L. Wang, X. Li, Y.-Q. Zhang, Y. Zhang, and K. Zhang. Evolution of scaling emergence in large-scale spatial epidemic spreading. *PloS one*, 6(7):e21197, 2011.

[124] Z. Wang, C. Wang, J. Pei, and X. Ye. Multiple source detection without knowing the underlying propagation model. In *Proc. of the 31st AAAI Conference on Artificial Intelligence*, pages 217–223. AAAI Press, 2017.

[125] L. Wasserman. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media, 2013.

[126] J. Wolf. Born again group testing: Multiaccess communications. *IEEE Transactions on Information Theory*, 31(2):185–191, 1985.

[127] Y. Wu and S. Verdú. Rényi information dimension: Fundamental limits of almost lossless analog compression. *IEEE Transactions on Information Theory*, 56(8):3721–3748, 2010.

[128] J. Yorke and H. Hethcote. *Gonorrhea: Transmission dynamics and control. Lecture notes in Biomathematics, 56, 1–105.* Springer-Verlag, Berlin, 1984.

[129] L. Zdeborová and F. Krzakala. Statistical physics of inference: Thresholds and algorithms. *Advances in Physics*, 65(5):453–552, 2016.

[130] J. Zhao, J. Wu, X. Feng, H. Xiong, and K. Xu. Information propagation in online social networks: a tie-strength perspective. *Knowledge and Information Systems*, 32(3):589–608, 2012.

[131] K. Zhu, Z. Chen, and L. Ying. Catch'em all: Locating multiple diffusion sources in networks with partial observations. In *Proc. of the 31st AAAI Conference on Artificial Intelligence*, pages 1676–1683. AAAI Press, 2017.

[132] K. Zhu and L. Ying. Information source detection in networks: Possibility and impossibility results. In *Proc. of 35th IEEE International Conference on Computer Communications, INFOCOM*, pages 1–9. IEEE, 2016.

Appendix A

Contained Publications and Contribution

Since all of the contained work are joint contributions, this chapter will provide a short overview of the authors' (LK) main contributions for each work as well as the corresponding publication status. The authors are abbreviated by their initials. The arXiv versions of the papers are attached at the end.

A.1 Noisy Group Testing via Spatial Coupling

This manuscript by Amin Coja-Oghlan, Max Hahn-Klimroth, Lukas Hintze, Dominik Kaaser, Lena Krieg, Maurice Rolvien and Olga Scheftelowitsch appeared in Combinatorics, Probability and Computing (CPC) 2024 [33]. This work on noisy group testing in the sublinear regime gives general lower bounds for approximate recovery as well as lower bounds for the constant column design for exact recover on the tests necessary. Additionally it provides efficient algorithms for both cases as matching upper bounds for both cases.

The overall proof strategy was jointly contributed by ACO, LH, LK, MR and OS. LK and OS jointly proved [33, Lemma 3.2, Lemma 3.10]. ACO, LK and OS carried out the Belief Propagation ansatz. LK proved that the result coincides with [25] for the special case of symmetric noise.

A.2 Noisy Linear Group Testing: Exact Thresholds and Efficient Algorithms

This manuscript by Lukas Hintze, Lena Krieg, Olga Scheftelowitsch and Haodong Zhu is accepted for the 38th Annual Conference on Learning Theory (COLT 2025) [65]. This project was initialised on a workshop in Strobl 2023 where the main strategies for all Theorems where discussed jointly. It provides lower bounds for noisy group testing in the linear regime in both the non adaptive and the adaptive case as well as efficient algorithm as upper bounds on the number of tests necessary for exact recovery.

The detailed strategy for for all proofs were later jointly discussed with LH, OS, HZ and LK, the detailed proof of Theorem 4 (Theorem 8 in Section 3.5, [65, Theorem 4]) was contributed by LK.

A.3 The k-XORSAT threshold revisited

This manuscript by Amin Coja-Oghlan, Mihyun Kang, Lena Krieg and Maurice Rolvien appeared in The Electronic Journal of Combinatorics (EJC), Volume 31 and as a short version in the Proceedings of EuroComb 2023 [34]. The work provides a proof of the random k-XORSAT satisfiability threshold theorem. The proof involves techniques from statistical physics combined with a specific moment computation. As a further result, the authors obtain the full rank threshold for sparse random matrices over finite fields with precisely k non-zero entries per row.

LK and MR worked on counting the Warning Propagation fixed points. MR worked on the lemma about the statistics of the vectors in the kernel of the matrix $\boldsymbol{A}^{\dagger}$. ACO carried out the generalization to

matrices over finite fields. ACO, LK, and MR worked on the adjustment of the pinning to the XOR-SAT problem. Applying the interpolation method was discussed jointly.

A.4 Inference of a Rumor's Source in the Independent Cascade Model

This manuscript by Petra Berenbrink, Max Hahn-Klimroth, Dominik Kaaser, Lena Krieg and Malin Rau appeared in 39th Conference on Uncertainty in Artificial Algorithms (UAI), 2023 [14]. This project was initiated by the ADYN research group. This work gives the precise condition on inferencing the source of a rumour in the independent cascade model for d-regular and Galton-Watson trees.

The formal details of the proofs were performed by MHK and MR. The critical property of the studied model necessary for the proofs was observed and contributed by LK.

Appendix B

Noisy Group Testing via Spatial Coupling

The following pages are left blank intentionally. The referenced article can be found at [33]. The arXiv version is available at `https://doi.org/10.48550/arXiv.2402.02895`.

Appendix C

Noisy Linear Group Testing: Exact Thresholds and Efficient Algorithms

The following pages are left blank intentionally. The referenced article can be found at [65]. The arXiv version is available at `https://doi.org/10.48550/arXiv.2411.03839`.

Appendix D

The k-XORSAT threshold revisited

The following pages are left blank intentionally. The referenced article can be found at [34]. The arXiv version is available at `https://doi.org/10.48550/arXiv.2301.09287`.

Appendix E

Inference of a Rumor's Source in the Independent Cascade Model

The following pages are left blank intentionally. The referenced article can be found at [14]. The arXiv version is available at `https://doi.org/10.48550/arXiv.2205.12125`.