Detection of locally stationary segments in time series – algorithms and applications

Uwe Ligges[1], Claus Weihs[1] and Petra Hasse-Becker[2]

[1] *Fachbereich Statistik*
Universität Dortmund
44221 Dortmund
Germany

[2] *Institut für Musik und ihre Didaktik*
Universität Dortmund
44221 Dortmund
Germany

Abstract. In many applications it is required to segment a time series into its locally stationary parts. Two applications are presented:

As a first example consider online monitoring of a BTA Deep-Hole-Drilling process. Here chattering and spiralling of the drilling tool should be avoided by process control. A second example is the analysis of vocal sound signals. It may be required to analyze only specific tones instead of a whole song.

Three new algorithms are introduced in this paper, all based on the theory of Dahlhaus (1997) and the analysis of the spectrum of time series, but with different methods to distinguish locally stationary parts of the signals.

1 Introduction

In many applications it is required to segment a time series into its locally stationary parts (cp. Dahlhaus (1997)), since on the one hand stationarity may be an assumption for further time series analysis, while on the other hand the transition areas may be of interest as in engineering or other technical applications.

Two quite different applications are presented:

As a first example consider online monitoring of a BTA (Boring and Trepanning Association) Deep-Hole-Drilling process (cp. Weinert et al. (2001)). Here chattering and spiralling of the drilling tool should be avoided by process control. This is possible by online monitoring of the process by analyzing locally stationary parts of the signal of the drilling torque. The transition between the occurance of changes in the time series and the start of chattering with an increased amplitude of the drilling torque is short, i.e. less than a second, so a fast algorithm has to be used.

A second example is the analysis of vocal sound signals. It may be required to analyse only specific tones instead of a whole song, e.g. for pitch tracking or conversion from wave-format into midi-format. Manual segmentation of the tones of a whole song is very time consuming, so automatical detection of locally stationary parts of the wave is desirable.

In the two presented applications it is an obvious idea to use frequency analysis, for which the Fast Fourier Transform (FFT; cp. Brockwell and Davis (1991)) is a common method to calculate a periodogram. The basic ideas of the paper are:

- Analysis of segmentation by already known algorithms for detection of locally stationary segments in time series, particularly by the one introduced by Adak (1998).
- Development of new fast algorithms combining well known methods of time series analysis with fundamental facts of the area of application and with measures which can be calculated quickly.

Correspondingly, three new algorithms are introduced in this paper, all based on the theory of Dahlhaus (1997) and the analysis of the spectrum of time series, but with different methods to distinguish locally stationary parts of the signals. In these methods segmentation is obtained using

1. Kolmogorov-Smirnov distance of empirical spectral distributions being a basic approach applicable to many different problems,
2. halftone distance derived from estimated fundamental frequencies (cp. Weihs et al. (2001)), and
3. note classification by fundamental frequencies.

For the segmentation of vocal sound signals as a particular musical application, the algorithms are compared on artificial series of tones as well as on waves from real singing (cp. chapter 4). For more details related to these real singers experiments cp. Weihs et al. (2001), where we also describe first steps to find objective criteria for the assessment of the quality of vocal performance. Notice that in this paper the terms "note" are used for the graphical sign and the corresponding ideal musical sound planned by the composer, and "sung note" or "tone" for the realized audio event corresponding to a note.

For the BTA Deep-Hole-Drilling process the Kolmogorov-Smirnov method yields sufficing results, i.e. 'early' chattering and spiralling prediction. For details regarding the time series the analysis is based upon, cp. Weinert et al. (2001). Other methods applicable to monitoring the process, involving AR models or kernel densities, are described in Busse et al. (2001).

The paper is structured as follows. Chapter 2 provides the description of an exact determination of the fundamental frequency for vocal sound signals, which is required for exact note classification in section 3.4. In chapter 3 the algorithms are introduced, which are compared in chapter 4. A conclusion is given in chapter 5.

2 Exact determination of fundamental frequency for vocal sound signals

In all the segmentation algorithms that will be introduced in chapter 3, the Dahlhaus (1997) theory about piecewise local stationary time series is used and so periodograms are determined for parts of the time series. Assume the wave was sampled with 11kHz and the window for which the periodogram was calculated covers $n = 512$ observations. Then the values of the periodograms can be determined only for the following Fourier frequencies (in Hertz): $21.53, 43.07, 64.60, \ldots, 5512.50$. Therefore, since the difference between the frequencies corresponding to the very low notes E and F, e.g., is only 5 Hertz, tones corresponding to these two notes cannot be expected to be well separable, because the distance between two Fourier frequencies is roughly 21.5 Hertz. Assume that 10 tones per second are sung, then on the one hand the compared parts should be shorter than one tenth of a second. On the other hand, however, using sectors of roughly 0.05 seconds, roughly corresponding to 512 observations, would lead to problems when very low tones, e.g. corresponding to E and F, have to be separated.

Please notice that the observed periodogram in figure 1, which corresponds to the sine wave with 70 Hertz (dashed line), does not have the ideal form since 70 Hz is not a Fourier frequency. Obviously, a rough estimate of the basic frequency would lie at 64.6 Hz based on only the highest peak of the periodogram.

This "highest–peak" estimator can be improved, however, by "averaging" between the frequencies with non-zero periodogram value. Comparing mean, harmonical mean, geometrical mean and the following estimator (cp. Weihs et al. (2001)):

$$\hat{\lambda} = \lambda_h + \frac{\lambda_s - \lambda_h}{2} \left(\frac{v_s}{v_h} \right)^{\frac{1}{e}}, \tag{1}$$

which averages the frequency λ_h of the highest peak with value v_h and the frequency λ_s of the highest peak of the direct neighbors of λ_h with value v_s of the periodogram, the last estimator is the best one of these four in the following sense:

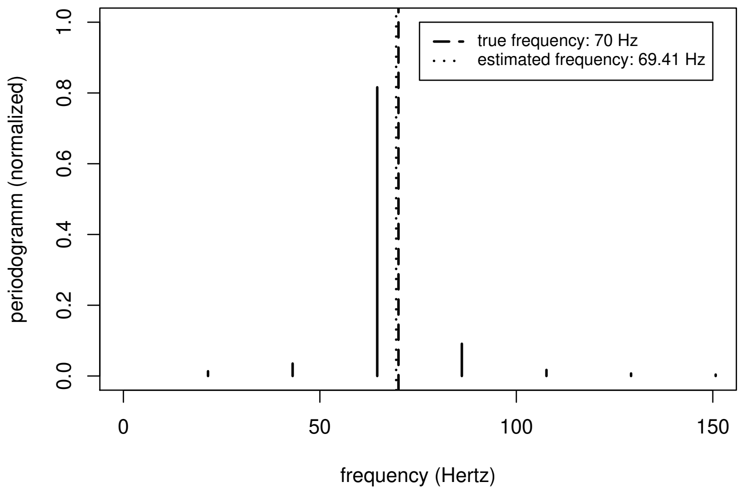

Figure 1: Periodogram – exact determination of fundamental frequency.

We simulated wave files (16-bit, 11kHz, 512 observations, sine waves without noise) for all halftones between D (73.4 Hz) and c'''' (2093 Hz)
(a) consisting of the fundamental frequency only and
(b) weighted with 70% of the fundamental frequency and 30% of the first overtone.
Then the maximum error of the approximation described in (1) was never larger than (a) 2.73 respective (b) 1.51 Hertz with a maximum MSE of (a) 0.36 respective (b) 0.29, while the maximal error of the other three "means" was never smaller than 5.06 Hertz with a minimum MSE of 7.53.

That means with the method described in (1) the true frequency can be estimated accurately enough to be able to separate even very low tones. In figure 1 the dotted line shows the frequency (69.41 Hz) estimated with this method.

3 Segmentation algorithms

Since the segmentation of vocal sound signals and online monitoring of the drilling torque (for BTA Deep-Hole-Drilling) are obviously special cases of change point detection, at first a look at already existing methods was taken. It can be assumed, that the expectation of the time series is equal to zero.

3.1 Algorithm by Adak

Since change from a musical point of view is not necessarily abrupt, but could also be somehow "smooth", as a first step the algorithm for "Time–Dependent Spectral Analysis of Nonstationary Time Series" introduced by Adak (1998) was implemented and used for segmentation of the "real singers" waves. This algorithm has been developed particularly for the segmentation of seismological data and speech, so hopefully it would also be useful for the segmentation of vocal sound.

A binary tree structure is used in this algorithm, which causes time expensive calculations, because on all levels of the tree, a FFT is calculated and particularly in the root of the tree a FFT of the whole time series has to be calculated. The computation time would be multiplied, if cross validation would be used as recommended by Adak (1998).

Even worse, the algorithm results in inexactness of segmentation if the time series being analyzed are "large". One cause, why this happens, is that change points detected in higher levels of the binary tree cannot be determined exactly as the parts of the time series represented by parts in the higher levels of the tree are quite large. A large number of both kinds of errors can be found: Detected change points in the middle of tones and not detected change points between different tones. Some error rates and comparisons to the other algorithms are shown in chapter 4. This leads to the conclusion that this is not an appropriate algorithm for the segmentation of vocal sound signals.

In the case of online monitoring of a BTA Deep-Hole-Drilling process, on the one hand the tree cannot be build every time, because future data is not available, on the other hand using this algorithm online, computation in time is not possible with recent computers.

Because the binary tree structure is the most problematic fact of the algorithm, in section 3.2 a new algorithm will be developed without the tree structure, nevertheless taking interesting ideas, like comparing cumulated periodograms using a KS–distance, from Adak's algorithm.

3.2 Segmentation using KS–distance

Since there were many problems with the tree structure in the algorithms by Adak (cp. section 3.1), we pass through the time series with a window of span n, for each window calculating the periodogram.

To compare two periodograms, the Kolmogorov–Smirnov–distance D^{KS} of their empirical spectral distributions is used. In principle, if this distance is larger than a threshold $w \in [0, 1]$, a change point is detected. For different singers different values of D^{KS} indicate a change point, e.g. because of different importance of overtones. So it is necessary to adjust the threshold w for each vocal times series.

If we know that the real number of tones of the song we want to separate is, say, $T_r \in \mathbb{N}$, the threshold w can be adjusted automatically so that approximately T_r segments will be found. The limits of the possible values of the threshold w can be set to reasonable values $w_{\min}, w_{\max} \in [0, 1]$ to speed up the algorithm. Now the "best" w can be determined by searching a number of detected change points T_d close to T_r on the following grid of possible thresholds: $w_{\max}, w_{\max} - g, w_{\max} - 2 \cdot g, \ldots, w_{\min}$ with $g \in [0, 1]$. Since T_d rises monotonously for smaller values of w, the search on the grid can be stopped, if $T_d > T_r$.

Let N be the length of the time series with values t_i $(i = 1, \ldots, N)$ and $n \in \{2^{\mathbb{N}}\}$ the size of a window, which passes through the time series. Further let us define a "tone" as a series of minimum l neighboring parts of the time series with size n. At the beginning of the algorithm, set $w := w_{\max}$:

1. Normalize, so that $t_i \in [-1, 1]$ $\forall i = 1, \ldots, N$ and $\bar{t} = 0$.
2. Set the number of parts to $B := \lfloor \frac{N}{n} \rfloor$.
3. For all $b = 1, \ldots, B$ estimate the empirical spectral distribution $\hat{S}_b$.
4. For all $b = 3, \ldots, (B-1)$ check as in (*), whether a change point has to be announced at the beginning of the b-th part.
5. Find whole "tones" in the segmented parts ("tone":= l neighboring parts).
   Set $T_d :=$ number of segmented tones.
6. If $((T_d < T_r) \wedge (w > w_{\min}))$ is true, replace w by $w - g$ $(w \leftarrow w - g)$ and go back to step 4.
7. Result:
   All detected change points (e.g. i-th point in the time series), start- and endpoints of tones, but cutting off $\frac{n}{2}$ points at each side of the tone.

(*) To avoid the segmentation of any "silence" in the wave because of strange effects of noise, a minimal variance is required. So parts of the time series with a variance smaller than a threshold $u \in \mathbb{R}$ are assumed to be "silence". A change point at the beginning of the b-th part of the time series will be detected, if and only if the variance of the b-th or the $(b-1)$-th part is larger than u and the following three inequalities are valid:

$$\begin{aligned} D^{KS}(\hat{S}_{b-1}, \hat{S}_b) &> w \quad \text{and} \\ D^{KS}(\hat{S}_{b-2}, \hat{S}_b) &> w \quad \text{and} \\ D^{KS}(\hat{S}_{b-1}, \hat{S}_{b+1}) &> w\,, \end{aligned} \tag{2}$$

where $D^{KS}(\hat{S}_{b-1}, \hat{S}_b)$ is the Kolmogorov–Smirnov–distance of the empirical spectral distributions $\hat{S}_{b-1}$ and $\hat{S}_b$. The three inequalities are required to make the algorithm more robust against vibrato. For theory regarding vocal performances (e.g. definition of "vibrato") compare Seidner and Wendler (1997).

Of course the algorithm sometimes detects change points at such locations, we want no change points to be detected, particularly if the singer is performing a strong vibrato on a note. Because of this, it happens that T_d is larger than T_r, even if not all change points are detected. Some experiments have shown heuristically that it

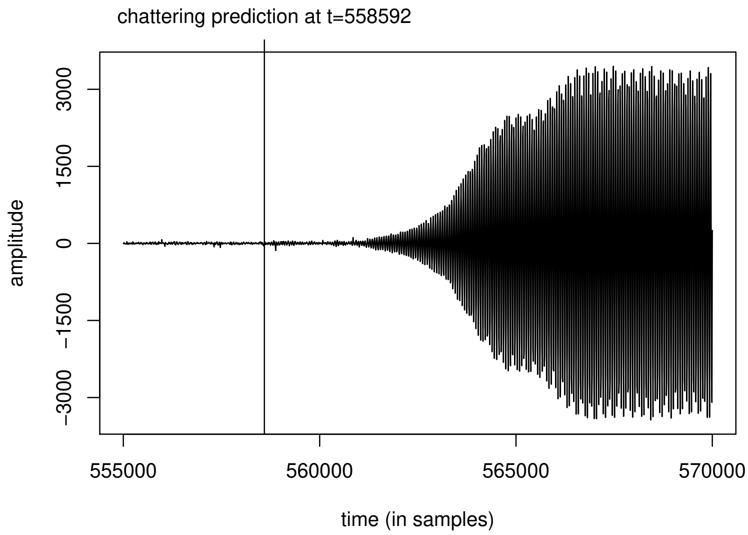

Figure 2: Chattering prediction on the drilling torque of a BTA-Deep-Hole-Drilling process.

is reasonable to raise T_r about 10 percent over its original value at the beginning of the algorithm.

It is possible to improve the accuracy: The windows passing through the time series are visiting neighboring parts (intervals) without overlapping. The above described algorithm can easily be changed so that the parts overlap each other, e.g. by $\frac{n}{2}$ points. In the following discussion the algorithm will be extended to allow for overlapping parts (intervals) which are compared for change point detection and a raised T_r by 10 percent.

The algorithm described in this chapter is much better in segmentation of vocal musical sound signals than the algorithm of Adak (a comparison in chapter 4). Nevertheless it is not really satisfying because of error rates above 20 percent on real singers data.

In figure 2 it is shown, however, that the algorithm is already satisfactory for online monitoring of the drilling process, because prediction of chattering is very early. Thus, no further algorithm was developed for this example.

3.3 Segmentation using halftone distance

Let us now change the algorithm described in section 3.2 in an attempt to solve the problem from a musical point of view. In the last algorithm, the Kolmogorov–Smirnov–distance D^{KS}, which is a well known and easy to calculate distance, was used. It can be replaced by the halftone distance D^{HT}, which is an implicitly given musical distance measure. The difference in halftones (D^{HT}) between the frequencies λ_1 and λ_2 is described by the following function (cp. Berg and Stork (1982)):

$$D^{HT}(\lambda_1, \lambda_2) := 12 \cdot log_2 \left(\frac{\lambda_1}{\lambda_2} \right). \tag{3}$$

In principle, to detect a change point, the distance in halftones between the fundamental frequencies of two neighboring parts of the time series must be equal or larger than one. In chapter 2 a method to estimate the fundamental frequency accurately enough is described. Since the halftone distance is a constant, in principle no threshold has to be adjusted like in the "KS–algorithm".

As in section 3.2 let N be the length of the time series with values t_i $(i = 1, \dots, N)$ and $n \in \{2^{\mathbb{N}}\}$ the span of a window, which passes through the time series. Further let us define a "tone" as a series of minimal l neighboring parts of the time series with size n. Then the algorithm is defined as follows:

1. Normalize, so that $t_i \in [-1, 1]$ $\forall i = 1, \dots, N$ and $\bar{t} = 0$.
2. Set the number of parts to $B := \lfloor \frac{N}{n} \rfloor$.
3. For all $b = 1, \dots, B$ estimate the fundamental frequency λ_b (chapter 2).
4. For all $b = 3, \dots, (B-1)$ check as in (**), whether a change point has to be announced at the beginning of the b-th part.
5. Find whole "tones" in the segmented parts ("tone":= l neighboring parts).
6. Result:
   All detected change points (e.g. i-th point in the time series), start- and endpoints of tones, but cutting off $\frac{n}{2}$ points at each side of the tone.

(**) As in section 3.2, parts of the time series with a variance smaller than a threshold $u \in \mathbb{R}$ are assumed to be "silence". A change point at the beginning of the b-th part of the time series will be detected, if and only if the variance of the b-th or the $(b-1)$-th part is larger than u and the following three equations are valid:

$$\begin{aligned} |D^{HT}(\lambda_{b-1}, \lambda_b)| &> 0.9 \quad \text{and} \\ |D^{HT}(\lambda_{b-2}, \lambda_b)| &> 0.9 \quad \text{and} \\ |D^{HT}(\lambda_{b-1}, \lambda_{b+1})| &> 0.9 \,. \end{aligned} \tag{4}$$

On the one hand these three inequalities are required to make the algorithm more robust against vibrato, on the other hand many singers are sliding from one tone to the other, so the "absolute" difference of 1 between two halftone is being reduced heuristically to 0.9, which was a good optimization in some experiments.

The error rate of the algorithm described in this chapter was in first comparisons (cp. chapter 4) approximately as good (or bad) as the error rate of the "KS–algorithm" described in section 3.2.

3.4 Segmentation using note classification by fundamental frequency

Another measure to distinguish tones, and so to segment the time series, is the note a tone corresponds to. Under the assumption that the frequency tuning of instruments is known, e.g. diapason $a' = 440$ Hertz, it is possible to perform a classification. For each part of the vocal time series the fundamental frequency can be estimated from the periodogram as already described. After that, this part can be classified by the fundamental frequency into its class: a particular note. Given λ_0 is the (known) frequency of a reference note, e.g. $a' = 440$ Hertz, this can be achieved by using the fundamental frequency note classifier

$$C^{FF}_{\lambda_0}(\lambda) := \left\lfloor 12 \cdot log_2 \left(\frac{\lambda}{\lambda_0} \right) + \frac{1}{2} \right\rfloor , \tag{5}$$

which is derived straightforward from formula (3), for a fundamental frequency λ.

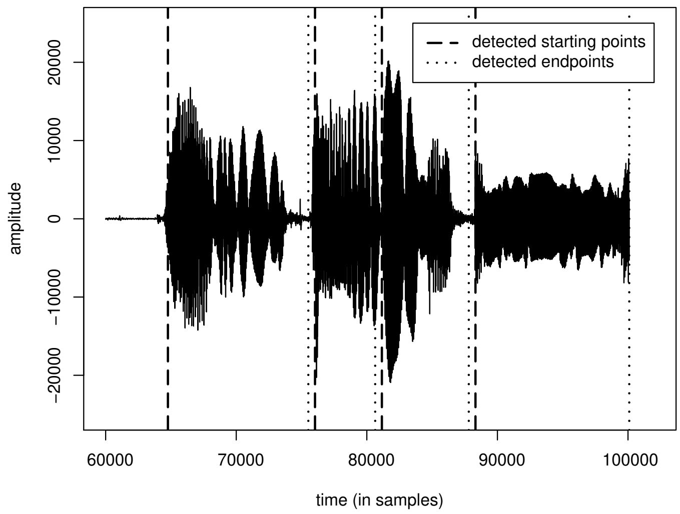

Figure 3: Fragment of a segmented vocal sound signal.

All in all, this algorithm works analogously to that described in section 3.3, except the three equations labelled by (4) have to be replaced by:

$$\begin{aligned} |C^{FF}_{\lambda_0}(\lambda_{b-1}) - C^{FF}_{\lambda_0}(\lambda_b)| &> 0 \quad \text{and} \\ |C^{FF}_{\lambda_0}(\lambda_{b-2}) - C^{FF}_{\lambda_0}(\lambda_b)| &> 0 \quad \text{and} \\ |C^{FF}_{\lambda_0}(\lambda_{b-1}) - C^{FF}_{\lambda_0}(\lambda_{b+1})| &> 0 \,. \end{aligned} \tag{6}$$

In the comparisons (chapter 4) it is shown, that on real singers' vocal sound signals this algorithm has the best segmentation rate of all algorithms described here. A typical segmentation of a fragment of the vocal sound signal from a semi-professional bass singer performing "Tochter Zion" is shown in figure 3. In this particular fragment the algorithm has segmented the time series very accurately.

4 Comparison of the algorithms

After the algorithms described in chapter 3 were implemented with the statistical software "R" (cp. Ihaka and Gentleman (1996)) three kinds of comparisons were done.

As already mentioned in section 3.1, the algorithm by Adak is neither appropriate for the segmentation of vocal sound signals nor for the chattering prediction of a BTA Deep-Hole-Drilling process, because it has a large runtime and it produces inaccurate results, which is obvious when looking at the results in section 4.1. So in the last two comparisons this algorithm is not compared to the others.

4.1 Simulation: artificial series of tones

In a first test of the algorithms two waves (16 bit, 11kHz) of each 25 tones are artificially generated and the length of the tones is randomly chosen between 0.05 and 1 second with abrupt change points. Also the pitch is randomly chosen (from D to f'''), except tones 7 - 12 with fixed pitch: D, Dis, d', dis', d''', dis'''; in order to test the algorithms on neighboring halftones. The "only" difference between the two generated waves is the weighting of fundamental frequency and overtones (sine–waves), which is set as follows:
(a) fundamental frequency 70%, 1$^{\text{st}}$ overtone 20%, 2$^{\text{nd}}$ overtone 10% and
(b) fundamental frequency 15%, 1$^{\text{st}}$ overtone 70%, 2$^{\text{nd}}$ overtone 15%.
The parameter n (span of the window) was set to 512 for all algorithms.

The results of the segmentation are shown in table 1. In the first columns of that table you will find: name of the note, corresponding frequency (Hertz), change point at the end of the note (in samples·100). In the following columns the results of the segmentation procedure of all four algorithms on both waves are shown. More than one number per line means the algorithm has incorrectly detected more than one change point during the tone. "NA" means, the algorithm has not detected a change point. Errors are printed in bold font type. "Error" means the particular algorithm did not detect a change point or the algorithm detects the change point inaccurately

note	frequ.	change p.	Adak a	Adak b	KS a	KS b	HT a	HT b	NC a	NC b
e	164.8	17	**NA**	**NA**	15	15	18	18	20	20
d	146.8	24	**NA**	**NA**	26	26	26	26	26	26
fis	185.0	66	**NA**	**NA**	67	67	67	67	67	67
c'	261.6	120	**NA**	**NA**	120	120	120	123	118	123
f	174.6	203	**163** 203	**163** 203	205	205	**154** **179** 200	**195** 205	**210**	**210**
g	196.0	210	**NA**	**NA**	210	210	210	215	215	215
D	73.4	222	**NA**	**NA**	220	**NA**	**NA**	230	225	**NA**
Dis	77.8	255	**NA**	**NA**	256	256	256	256	256	256
d'	293.7	331	**NA**	**NA**	**NA**	327	328	333	333	333
dis'	311.1	425	429	429	425	425	425	430	430	430
d'''	1174.7	513	511	**450** **490** 511	515	515	515	517	517	512
dis'''	1244.5	532	531	531	532	532	532	532	532	532
g	196.0	624	**572** 622	**572** **612**	625	625	625	630	630	630
f'	349.2	635	**645**	**622**	635	635	635	635	635	635
a'	440.0	708	**716**	**NA**	707	707	712	712	712	712
cis''	554.4	755	**797**	**797**	753	753	758	753	753	753
c'	261.6	865	**878**	**878**	865	865	865	865	865	865
b	233.1	915	919	919	916	916	916	916	911	911
fis	185.0	973	**NA**	**962**	973	973	978	973	973	978
cis	138.6	992	**1003**	**NA**	988	988	993	998	993	998
f'	349.2	1029	**1043**	**NA**	1029	1029	1034	1034	1034	1034
dis''	622.3	1084	1084	**NA**	1080	1080	1085	1085	1085	1085
f''	698.5	1192	**1125**	**NA**	1193	1193	1188	1193	1193	1193
cis'	277.2	1271	**NA**	**NA**	1270	1270	**NA**	**NA**	**NA**	**NA**
d''	587.3	1290	1290	1290	1290	1290	1290	1290	1290	1290
errors			19	23	1	1	4	2	2	3

Table 1: Comparison with an artificial series of tones (change points in samples·100).

(i.e. more than 512 data points away from the real one) or the algorithm detects a change point at a place without any real change points.

A short description of the table headers follows:

`Adak`	algorithm by Adak (cp. section 3.1),
`KS`	segmentation algorithm using KS–distance (section 3.2),
`HT`	segmentation algorithm using halftone distance (section 3.3),
`NC`	segmentation algorithm using note classification (section 3.4),
`a`	for the wave with frequency weighting 70%, 20%, 10% (see above),
`b`	for the wave with frequency weighting 15%, 70%, 15% (see above).

Example for the interpretation of table 1
The `KS`–algorithm produces one error for each wave at the halftone change points $D - Dis$ (wave `b`) and $d' - dis'$ (wave `a`), because it does not detect the change point. The reason is that the pitch, and therefore the frequency, changes only slightly and the shift probably cannot be distinguished from a vibrato.

Since there are no large differences in the segmentation exactness of the algorithms `KS`, `HT` and `NC` for this problem (minimal one error, maximal four errors; cp. table 1), more comparisons have to be done.

4.2 Simulation: An artificial performance of "Tochter Zion"

In a second step artificial $4 \cdot 252 = 1008$ waves of "Tochter Zion" were generated, each sampled with 11 kHz and 16 bit and consisting of 77 tones. The duration of a half note corresponds to 1 sec and between two notes there are breaks of $\frac{1}{100}$ seconds duration. Four versions are required for the different types of singing voices (soprano, alto, tenor, bass). For each of these versions an experimental design is used to simulate different types of voices related to the weighting of fundamental frequency and its first five corresponding overtones. All of these six frequencies are weighted by all of the following ratios $\frac{0}{6}, \frac{1}{6}, \ldots, \frac{6}{6}$ with the restrictions that the fundamental frequency must have at least a weighting of $\frac{1}{6}$ and that the sum of the weights must be equal to one. So this design results in 1008 waves totally

	soprano			alto			tenor			bass		
	KS	HT	NC	KS	HT	NC	KS	HT	NC	KS	HT	NC
Min.	8.00	8.00	8.00	8.00	6.00	5.00	8.00	3.00	6.00	8.00	3.00	3.00
Median	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	13.00	8.00
Mean	8.01	8.41	8.00	8.01	7.77	7.41	8.03	7.56	7.94	8.04	12.66	8.51
Max.	11.00	9.00	8.00	11.00	9.00	8.00	10.00	10.00	10.00	11.00	23.00	16.00

Table 2: Error rates of a segmentation of artificially generated waves of "Tochter Zion".

(252 permutations of overtone weighting multiplied by the four versions mentioned above).

In table 2 statistics about error rates corresponding to the four types of singing voices are shown for the algorithms `KS, HT, NC` (cp. section 4.1). The number 8 appears at many places in table 2, because 8 times there are two neighboring notes with the same pitch only interrupted by breaks of $\frac{1}{100}$ seconds duration. Obviously these tones are problematic to be distinguished, because the corresponding periodograms are similar, nevertheless sometimes these "change points" are detected astonishingly.

Looking at the error rates in table 2, the `KS`–algorithm seems to be the most robust one (particularly for bass singers), while the other two algorithms in some circumstances have very good minimal error rates.

4.3 Segmentation on real singers' performances

Since in the simulation described in section 4.2 no noise and no vibrato was added to the artificially generated sine waves, we finally use real data to find out, whether the algorithms work on real vocal sound signals appropriately.

So at last the algorithms are compared by results of segmentations of 17 versions of the classical song "Tochter Zion" (Händel) performed by 17 singers (from real amateurs to real professionals) to a standardized piano accompaniment played back by headphones. The waves were recorded in CD quality, that means with sampling

	soprano				alto						tenor			bass				sum
	1	2	3	4	1	2	3	4	5	6	1	2	3	1	2	3	4	
KS	35	20	31	38	25	18	31	22	26	28	27	40	35	31	24	31	34	496
HT	49	24	32	36	26	29	26	25	19	24	39	28	23	28	27	32	33	500
NC	27	19	20	25	22	18	12	9	6	12	12	17	16	27	22	21	23	308

Table 3: Error rates of 17 real singers performances of "Tochter Zion".

rate 44100 Hertz in 16-bit format. For time series analysis the waves were reduced to 11kHz (in order to restrict the number of data), and standardized to the interval [-1,1]. For more details related to these real singers' experiments cp. Weihs et al. (2001), where we also describe first steps to find objective criteria for the assessment of the quality of vocal performance.

The number of errors in this experiment is shown in table 3, where each column (2-18) represents one particular singer. Obviously the segmentation algorithm using note classification is much better (error rate $< 20\%$) than the error rate of the other algorithms (roughly 30%) on real singers performances.

5 Conclusion

In section 3.2 it is shown that the algorithm "Segmentation using KS–distance" already satisfies for online monitoring of a BTA Deep-Hole-Drilling process, because prediction of chattering is possible early.

From the musical point of view the cause of the most errors of the newly developed algorithms is the appearance of vibrato which can be observed frequently and in a strong manner in professional singers' performances. Even in one particular case, one long tone sung by a professional bass singer was divided into eight segments. Vibrato is not or only slightly observed on time series derived from vocal sound signals of singing amateurs. For these time series the error rate of the segmentation algorithms is much lower. Applying the algorithms on pure sine waves as in sections 4.1 and 4.2 shows that without vibrato an accurate segmentation is possible.

The algorithms "Segmentation using KS–distance" (cp. section 3.2) and "Segmentation using halftone distance" (cp. section 3.3) are quite equal in their error rates and computational time consumption. The algorithm "Segmentation using note classification by fundamental frequency" (cp. section 3.4) is the best of the here described algorithms on segmentation of real singers' vocal performances with an error rate less than 20%.

Acknowledgements. The financial support of the Deutsche Forschungsgemeinschaft (SFB 475, "Reduction of complexity in multivariate data structures") is gratefully acknowledged.

References

[1] Adak, S. (1998). Time-Dependent Spectral Analysis of Nonstationary Time Series. *Journal of the American Statistical Association*, **93**, 1488-1501.

[2] Berg, R. E. and Stork, D. G. (1982). *The Physics of Sound.* New Jersey: Prentice-Hall.

[3] Brockwell, P. J. and Davis, R. A. (1991). *Time Series: Theory and Methods.* New York: Springer.

[4] Busse, A. M., Hüsken, M. and Stagge, P. (2001). Offline-Analyse eines BTA-Tiefbohrprozesses. *Technical Report 16/2001.* SFB 475, Department of Statistics, University of Dortmund, Germany.

[5] Dahlhaus, R. (1997). Fitting Time Series Models to Nonstationary Processes. *The Annals of Statistics 1997*, **25**, 1-37.

[6] Ihaka, R. and Gentleman, R. (1996). R: A Language for Data Analysis and Graphics. *Journal of Computational and Graphical Statistics*, **5**, 299-314.

[7] Seidner, W. and Wendler, J. (1997). *Die Sängerstimme.* Berlin: Henschel.

[8] Weihs, C., Berghoff, S., Hasse-Becker, P. and Ligges, U. (2001). Assessment of Purity of Intonation in Singing Presentations by Discriminant Analysis. In: Kunert, J. and Trenkler. G. (2001). *Mathematical Statistics and Biometrical Applications*, 395-410. Lohmar: Josef Eul Verlag.

[9] Weinert, K., Webber, O., Busse, A., Hüsken, M., Mehnen, J. and Stagge P. (2001). In die Tiefe: Koordinierter Einsatz von Sensorik und Statistik zur Analyse und Modellierung von BTA-Tiefbohrprozessen. In: Spur, G. (2001) *ZWF, Zeitschrift für wirtschaftlichen Fabrikbetrieb*, **5**. München: Carl Hanser Verlag.