Available online at www.sciencedirect.com
ScienceDirect
Speech Communication 48 (2006) 1650–1665
SPEECH COMMUNICATION
www.elsevier.com/locate/specom

# Poincaré pitch marks

Martin Hagmüller *, Gernot Kubin

*Signal Processing and Speech Communication Laboratory, Graz University of Technology, Inffeldgasse 12, 8010 Graz, Austria*

Received 14 March 2006; received in revised form 21 July 2006; accepted 26 July 2006

**Abstract**

A novel approach for pitch mark determination based on dynamical systems theory is presented. Pitch marks are used for speech analysis and modification, such as jitter measurement or time scale modification. The algorithm works in a pseudo-state space and calculates the Poincaré section at a chosen point in the state space. Pitch marks are then found at the crossing of the trajectories with the Poincaré plane of the initial point. The procedure is performed frame-wise to account for the changing dynamics of the speech production system. The system is intended for real-time use, so higher-level processing extending over more than one frame is not used. The processing delay is, therefore, limited to one frame. The algorithm is evaluated by calculating an average pitch value for 10 ms frames and using a small database with pitch measurements from a laryngograph signal. The results are compared to a reference correlation-based pitch mark algorithm. The performance of the proposed algorithm is comparable to the reference algorithm, but in contrast correctly follows the pitch marks of diplophonic voices.


*Keywords:* Poincaré section; Pitch marks; State space; Nonlinear speech processing; Diplophonic voice; Disordered voice; Jitter

## 1. Introduction

This work is motivated by the need for a pitch marking system, which can be applied on running speech and gives results in real-time. The application which it is needed for is a voice enhancement system for speakers with disordered voice (e.g., see Hagmüller and Kubin (2004)). For practical use in everyday life the processing delay has to be as low as possible. For example, if a potential voice augmentation device is used to enhance a telephone conversation, the processing delay of the device adds up with the delay already introduced by speech coders for the telephone channel. With this in mind the aim is a reliable algorithm which only needs a short frame buffer to keep the processing delay as low as possible.

With Poincaré sections we chose a promising approach which originates in nonlinear system theory. Since this field is rather new to the speech processing community this paper will present the necessary theoretical background and provides a step-by-step description of the algorithms to open up the field of nonlinear time series analysis to the reader not familiar with the subject.

The resulting pitch marking system is compared to the pitch marking system provided by the speech

* Corresponding author. Tel.: +43 316 873 4377; fax: +43 316 873 4432.
*E-mail address:* hagmueller@tugraz.at (M. Hagmüller).

0167-6393/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.specom.2006.07.008

processing software package 'Praat' by Boersma and Weenink (2005).

### 1.1. Applications of pitch marks

Pitch marks are essential for several speech processing methods. For speech modification, Moulines and Laroche (1995) proposed the time-domain pitch-synchronous overlap-and-add (TD-PSOLA) technique. It allows pitch modification of a given speech signal without changing the time duration and vice versa. Single pitch cycles are manipulated and, therefore, the pitch cycles have to be determined reliably. This method is widely used in concatenative speech synthesis to modify the stored speech segments according to the desired prosody (i.e., fundamental frequency ($F_0$) and duration).

Pitch-synchronous speech enhancement is another application, where pitch-marks are needed. Kleijn (2002) proposed a procedure based on averaging over neighboring pitch cycles and applied it to coded speech. It enhances deterministic harmonic components and reduces stochastic noise. Hagmüller and Kubin (2004) applied this speech enhancement method to disordered voice augmentation.

Pitch marks can also be used to determine an instantaneous frequency contour and to calculate cycle-based analysis measures such as jitter (Schoentgen, 2003). Based on such analysis, further applications such as intonation recognition (Nöth et al., 2002) for dialogue systems or voice disorder detection for health care are possible (Titze, 1994).

If a system has to be applied in a live communication setting, then a major requirement for the algorithm is that the results can be obtained in real time with minimal delay.

### 1.2. Nonlinear processing of speech

Linear methods have been applied successfully to speech processing problems and are widely accepted in the speech processing community. Not all phenomena occurring in human speech can be explained by linear models. In the particular case of disordered voices the limitations are clearly observable. Therefore, nonlinear approaches for speech processing have been receiving wider attention for just over a decade as limitations of linear models call for more general signal models.

With the emergence of nonlinear dynamical systems analysis, researchers started to apply low dimensional dynamical models to speech processing Tishby (1990). Specifically, the vocal fold oscillation has received a considerable amount of attention from the viewpoint of nonlinear dynamics (e.g., Herzel et al., 1995 and Giovanni et al., 1999a). Phenomena like bifurcations, subharmonics or period-doubling—as occur in diplophonic voice—and chaotic behavior have all been observed in the human voice. Human speech has been examined in terms of Lyapunov exponents and correlation dimensions, among others (Banbrook et al., 1996; Kumar and Mullick, 1996; Kokkinos and Maragos, 2005). For an overview of nonlinear speech processing, see Kubin (1995). From meetings dealing specially with nonlinear speech processing several publications resulted, which also provide an overview of the state-of-the-art in the field (Bimbot, 2003; Chollet et al., 2005; Faundez-Zanuy et al., 2006).

For disordered voices, nonlinear approaches have received considerable attention for analysis, in particular as an objective alternative to auditory voice evaluation methods (e.g., Giovanni et al., 1999b; Titze, 1994; Jiang et al., 2006; Little et al., 2006). More recently, state-space approaches have been used for noise reduction (e.g., Hegger et al., 2001, 2000; Matassini and Manfredi, 2002; Johnson et al., 2003). State-space methods have also been applied to improve automatic speech recognition (Indrebo et al., 2006).

This paper wants to introduce the value of specific state-space methods to the speech community, and make the methodology accessible to a wider audience.

### 1.3. Pitch marks–perception–harmonicity–periodicity

Depending on the application, one wants to analyze either the periodicity or the harmonicity of the signal (see Fig. 1). This is specially of interest for irregularities of the fundamental frequency, which can either be interpreted as an alternating fundamental period in the time domain, or as an additional subharmonic component in the frequency domain. If signal modification is implemented in the frequency domain, such as harmonic plus noise modeling by Stylianou et al. (1995), the necessary information is only captured if the smallest occurring harmonic is considered. For time-domain based signal modification approaches such as TD-PSOLA (Moulines and Laroche, 1995) or analysis of the vocal fold movement, the smallest measured period has to be captured to reconstruct the irregularities of the voice.

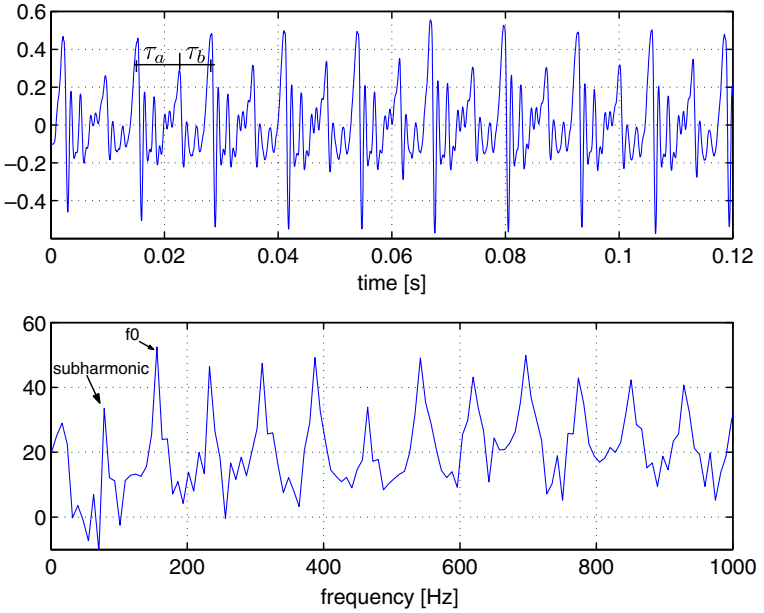


Fig. 1. Top: Wave-form view of a signal section (vowel 'o'). The two alternating fundamental periods, $\tau_a$ and $\tau_b$ can be analyzed. Bottom: Fourier transform of the signal. The subharmonic of the signal can be seen.

The perception of the subharmonic appearing in a speech signal depends on the relative energy contained in the subharmonic. In case of a weak subharmonic the perceived pitch stays the same, but a change in the vocal quality such as roughness occurs (Titze, 1994). When the subharmonic is strong, it is perceived as a new pitch.

## 2. Background and related work

Here we want to present the necessary background needed to understand the proposed algorithm. A more indepth coverage can be found in (Kantz and Schreiber, 2004).

### 2.1. Embedding of dynamical systems

If one wants to analyze a dynamical system the most efficient representation is the state space with an appropriate dimension. In continuous time the dynamics of a system are defined by $m$ first-order ordinary differential equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{s}(t) = \boldsymbol{f}(\boldsymbol{s}(t)), \quad t \in \mathbb{R}, \tag{1}$$

where $\boldsymbol{f}$ is a set of $m$ functions, $\boldsymbol{f}(\boldsymbol{s}(t)) = (f_1(\boldsymbol{s}(t)), f_2(\boldsymbol{s}(t)), \ldots, f_m(\boldsymbol{s}(t)))$ and $\boldsymbol{s}(t)$ is the $m$-dimensional state vector of the system. Eq. (1) is also called the *flow* of a system.

In discrete time the dynamics are described by a map $\boldsymbol{F}$

$$\boldsymbol{s}[n+1] = \boldsymbol{F}(\boldsymbol{s}[n]), \quad n \in \mathbb{Z}. \tag{2}$$

For both cases the state-space trajectory is specified by the temporal evolution of the state vector $\boldsymbol{s} \in \mathbb{R}^m$.

#### 2.1.1. Delay embedding

Human speech is usually available only as a one-dimensional signal, $x[n]$. Therefore, one has to convert this scalar signal into a state space representation. It has been shown that a nonlinear dynamical system can be embedded in a reconstructed state space by the method of delays (Kantz and Schreiber, 2004). The state space of a dynamical system can be topologically equivalently reconstructed from a single observed one-dimensional system variable (Takens, 1981).

An $M$-dimensional trajectory $\boldsymbol{x}(n)$ can be formed by delayed versions of the speech signal $x(n)$,

$$\boldsymbol{x}(n) = \{x[n], x[n-\tau_\mathrm{d}], \ldots, x[n-(M-1)\tau_\mathrm{d}]\}, \tag{3}$$

where $\tau_\mathrm{d}$ is the delay time, which has to be chosen so as to optimally unfold the attractor. An attractor is a bounded subset of the phase space onto which, after some transient time, the trajectories will have converged. Trajectories with initial points outside the attractor but within the 'basin of attraction' will be attracted to that subset of the phase space. This

subset can be a point, a curve, or a more complicated topological structure.

### 2.1.2. Embedding dimension

The optimal choice of the dimension for state space embedding is an important issue. If the dimension is too small the determinism of the signal flow is no longer preserved, on the other hand, if the dimension is chosen too large, the computational effort rises. For a $D$-dimensional attractor it is sufficient to form an $M \geqslant 2D+1$ state space vector (Takens, 1981). Later this result was generalized by Sauer et al. (1991) to $M > 2D_F$, where $D_F$ is the (fractal) box counting dimension of the attractor. In practice, though, smaller values of $M > D_F$ can be sufficient. The method of detecting false nearest neighbors can be used to determine the minimal necessary embedding dimension (Kantz and Schreiber, 2004).

### 2.1.3. Embedding delay

For the optimal delay, no mathematical formulation exists, only from a practical point of view, a goal is to optimally unfold the attractor. That means the extension of the attractor would be roughly the same in all dimensions. This is the case, when the reconstructed state vector components have minimal statistical dependence of each other. The most natural approach would be to use the autocorrelation function of the signal, which is a good choice, but is only based on linear statistics. Both linear and nonlinear dependencies can be calculated by the auto-mutual information, also known as the time-delayed mutual information. This is an information theoretic concept, which is based on the Shannon entropy. It computes the statistical dependencies between different time delays of a signal. The auto-mutual information for a time delay $\tau$ is defined as

$$I_\epsilon(\tau) = \sum_{i,j} p_{i,j}(\tau) \ln p_{i,j}(\tau) - 2\sum_i p_i \ln p_i, \qquad (4)$$

where $\epsilon$ is the resolution of the histogram estimate for the probability distribution of the data, $p_i$ is the probability that the signal has a value which lies in the $i$th bin of the histogram and $p_{i,j}$ is the probability that $s(t)$ is in bin $i$ and $s(t+\tau)$ is in bin $j$. The time lag $\tau$ at the first minimum of the auto-mutual information is the optimal delay time for the embedding. The resolution $\epsilon$ can be set rather coarse, since only the dependence of $I_\epsilon$ on $\tau$ is of interest and not the absolute value of $I_\epsilon(\tau)$ (Kantz and Schreiber, 2004).

### 2.1.4. Poincaré plane

If one chooses an arbitrary point on the attractor in an $M$-dimensional space then one can create a hyper-plane which is orthogonal to the flow of the trajectories at the chosen point. This is called the Poincaré plane (Fig. 2). All trajectories, that return to a certain neighborhood of the initial point, cross the hyperplane and can be represented by their intersection with this plane in $M-1$ dimensions compared to the original $M$-dimensional trajectory.

## 2.2. Pitch detection in state space

Kubin (1997) first suggested to use Poincaré sections for the determination of pitch marks and mentioned special applications for signals with irregular pitch period. Experiments showed very promising results for an example with vocal fry, where the pitch period doubles for some time. The pitch period was followed correctly.

Later Mann and McLaughlin (1998) further worked with Poincaré maps and applied them to epoch marking for speech signals, with glottal closure instants set as initial point. They again saw promising results, but reported the failure to resynchronize after, e.g., stochastic portions of speech.

More recently, Terez (2002) introduced another state space approach to pitch detection, using space–time separation histograms. Each pair of points on the trajectory in state space is separated by a spatial distance $r$ and a time distance $\Delta t$. One can draw a scatter plot of $\Delta t$ versus $r$ or, for every time distance, count the pairs within a certain

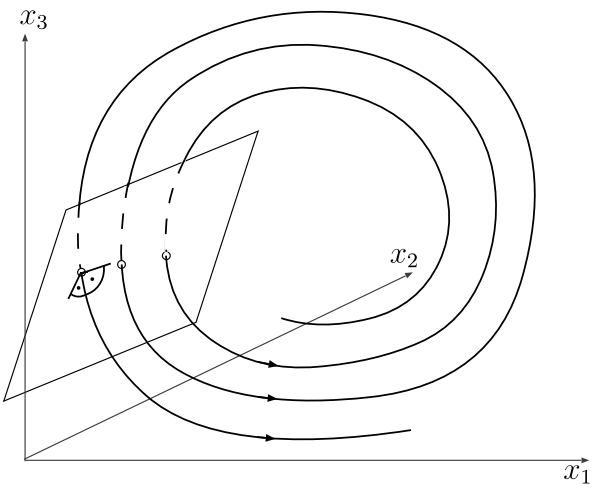


Fig. 2. Placement of the Poincaré plane orthogonal to the flow of the trajectories.

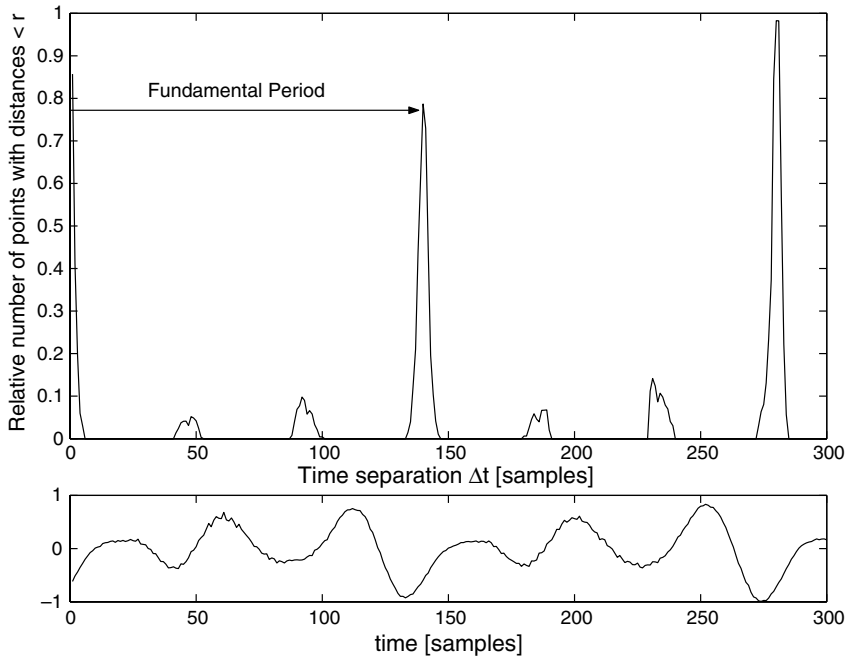


Fig. 3. Histogram of space–time separation. The normalized number of state-space distances within a certain neighborhood $r$ for every time distance $\Delta t$ is plotted.

neighborhood $r$. This count can then be normalized to 100% to yield a histogram (Fig. 3).

In case of periodicity in the signal, the histogram concentrates at certain $\Delta t$ values, whereas others have rather low values. The first maximum of the histogram indicates the fundamental pitch period. Compared to the auto-correlation function the peak is much more significant and, therefore, the author claims, it offers improved performance. In case of noise-like signals the histogram is more evenly spread over all time distances. Since histograms are based on averaging statistics, pitch marks cannot be determined reliably with this approach. The computed fundamental period is an average over the chosen frame length. Though, if the frame length is very short, the accuracy would be rather high.

## 3. Algorithm

This work builds on the aforementioned approaches and is an improved version of the work previously described in (Hagmüller and Kubin, 2003, 2005). A step-by-step guide is included in Matlab-like notation to allow a smooth and easy access to the area of nonlinear dynamical systems.

### 3.1. Pre-processing

The algorithm works on a frame-by-frame basis to handle the slowly changing parameters of the speech production system. For pitch mark detection, the low-dimensional characteristics of the signal need to be observed. So the noise has to be removed, otherwise, specially for hoarse voices, the attractor is hardly visible using three-dimensional embedding (Fig. 4). If the embedding dimension is high enough, intersections with the Poincaré plane would still correspond to the pitch period, but with less reliability. However, at a certain noise level the algorithm breaks down.

For a noise reduced attractor, a singular-value-decomposition (SVD) embedding approach has been proposed (Broomhead and King, 1986), but similar results can be achieved by a simple linear-phase low-pass filter (Fig. 5). The latter is computationally less demanding of course, so this is chosen here for noise reduction.

To remove the influence of a changing amplitude, automatic gain control or envelope smoothing is applied for every frame of the input signal $x_0[n]$. First, the signal envelope $v(n)$ is calculated

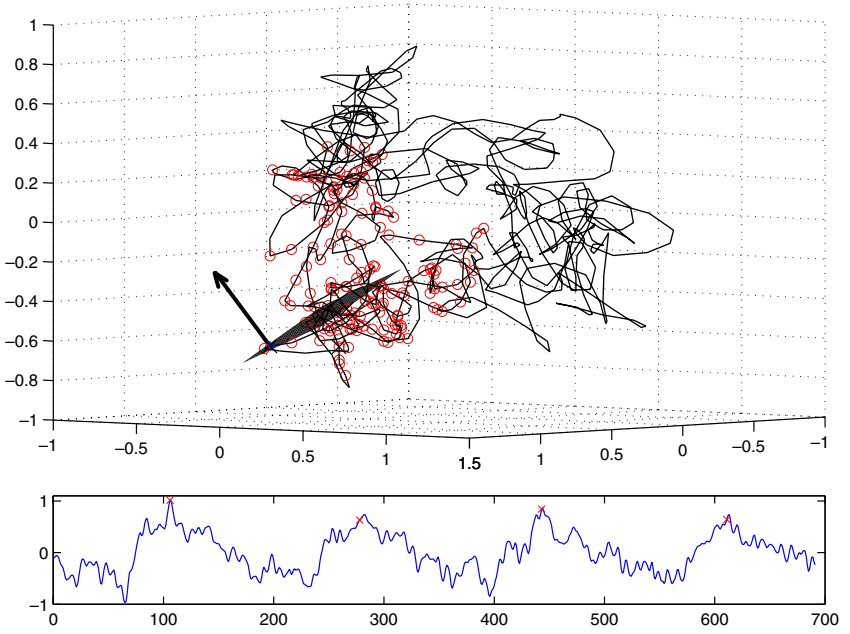

Fig. 4. Vowel 'a' from a dysphonic speaker. Top: Projection of state-space embedding on three-dimensions and Poincaré plane. Circles are neighbors. Bottom: Time-domain waveform plot. No low-pass filter.

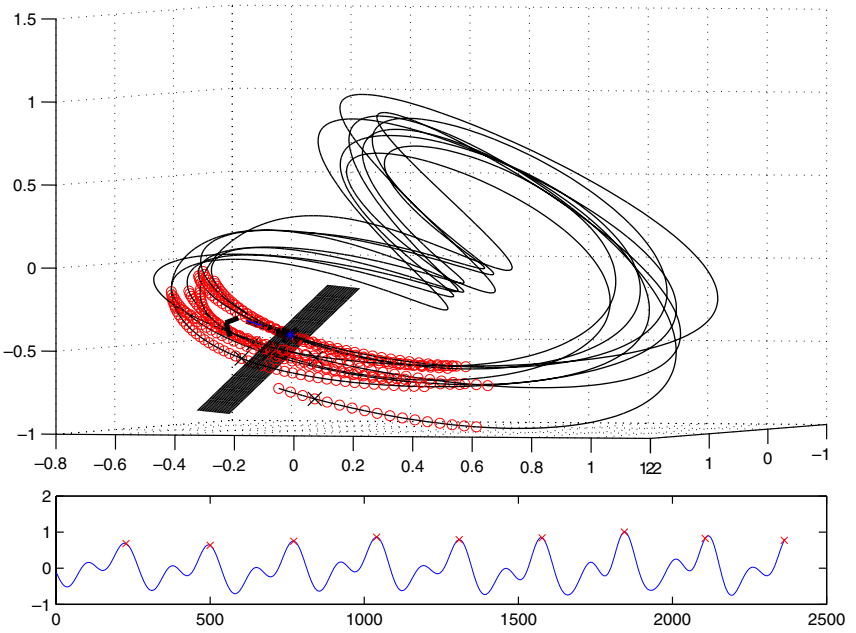

Fig. 5. Vowel 'a' from a dysphonic speaker. Top: Projection of state-space embedding on three-dimensions and Poincaré plane. Circles are neighbors. Bottom: Time-domain waveform plot. Low-pass filter.

$$v[n] = \frac{1}{K} \sum_{k=-\frac{K-1}{2}}^{\frac{K-1}{2}} |x_0[n+k]|, \tag{5}$$

$$x[n] = \frac{x_0[n]}{v[n]}, \quad \forall v[n] > v_{\text{th}}, \tag{6}$$

where $x[n]$ is the speech signal, $K$ is the length of the moving average filter, which is set to the maximum expected fundamental period and $v_{\text{th}}$ is a threshold to avoid overamplification of low-energy non-speech sections. This moves the trajectories of

quasi-periodic signals closer together, which means that the attractor is contracted, if it was spread due to amplitude variations (Fig. 6).

Then the signal is upsampled to $f_s = 48$ kHz to increase the resolution of the pitch marks, since at low sampling rates the pitch marks would exhibit too much discretization error. The embedding in state space is implemented by the method of delays, the embedding dimension is chosen to be $M = 8$. Experiments showed that this number gives the most robust results over different kinds of speech samples, though the algorithm will generally work with embedding dimensions $M \geqslant 3$. For every frame a state-space matrix $\boldsymbol{X}$ is formed

$$\boldsymbol{X} = \begin{pmatrix} x[0] & x[\tau_d] & \dots & x[(M-1)\tau_d] \\ x[1] & x[1-\tau_d] & \dots & x[1-(M-1)\tau_d] \\ \vdots & \vdots & \ddots & \vdots \\ x[N] & x[N-\tau_d] & \dots & x[N-(M-1)\tau_d] \end{pmatrix}, \tag{7}$$

where $N$ is the frame length, $M$ is the embedding dimension and $\tau_d$ is the embedding delay. Each row represents a point $\boldsymbol{x}(n)$ in state-space.

### 3.2. Poincaré section

At the heart of the algorithm is the calculation of the Poincaré hyperplane. First a point at time $n_0$ has to be chosen. This can either be a maximum or minimum in the time-domain waveform. A potential source of error is the choice of spurious peaks in the signal, not connected to the voice signal or the voice onset (see Fig. 13). Another possibility is to select the initial point directly in state space.

This method tries to find an area in state space, where the local bundle of trajectories is most parallel. The initial point is placed here. This provides for an improved reliability, since this is optimal for finding pitch marks using the Poincaré plane, and the time-domain wave form is not considered anymore.

Around this chosen query point $\boldsymbol{x}(n_0) = \boldsymbol{X}(n_0,:)$, the state space is searched for the $k$ closest points, according to the Euclidean distance measure forming a neighborhood $\mathcal{N}(n_0)$. This can be done by calculating the Euclidean distance between $\boldsymbol{x}(n_0)$ and all other points $\boldsymbol{x}(n)$ of the state-space matrix $\boldsymbol{X}$

$$d_{\text{eucl}}(n) = \sum_{m=1}^{M} (X(n,m) - X(n_0,m))^2. \tag{8}$$

There exist computationally efficient methods to search for the neighbors in state space (e.g., Schreiber (1995)).

Then a mean flow direction $\boldsymbol{f}(n_0)$ of the trajectories in this neighborhood $\mathcal{N}(n_0)$ is calculated

$$\boldsymbol{f}(n_0) = \text{mean}(\boldsymbol{x}(n+1) - \boldsymbol{x}(n)), \quad \forall n \in \mathcal{N}(n_0), \tag{9}$$

where only trajectories are considered, which point roughly in the same direction as the initial flow vector $\boldsymbol{f}_0^{T(n)} \boldsymbol{f}_0(n_0) > 0.6$, where $\boldsymbol{f}_0$ is a unit-length vector), i.e., orthogonal flow vectors or flow vectors in the opposite direction will not be considered.

So for every frame the Poincaré hyperplane is defined as the hyperplane through $\boldsymbol{x}(n_0)$, which is perpendicular to $\boldsymbol{f}(n_0)$ (Fig. 7(b)).

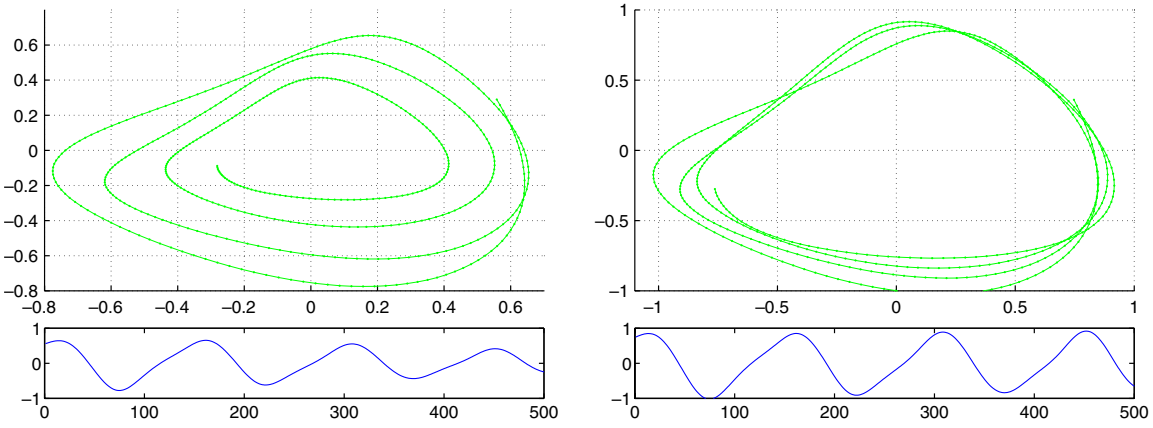


Fig. 6. Left: Projection of state-space embedding on two dimensions. Right: Time-domain waveform plot. Left: No automatic gain control applied. Right: Automatic gain control applied.

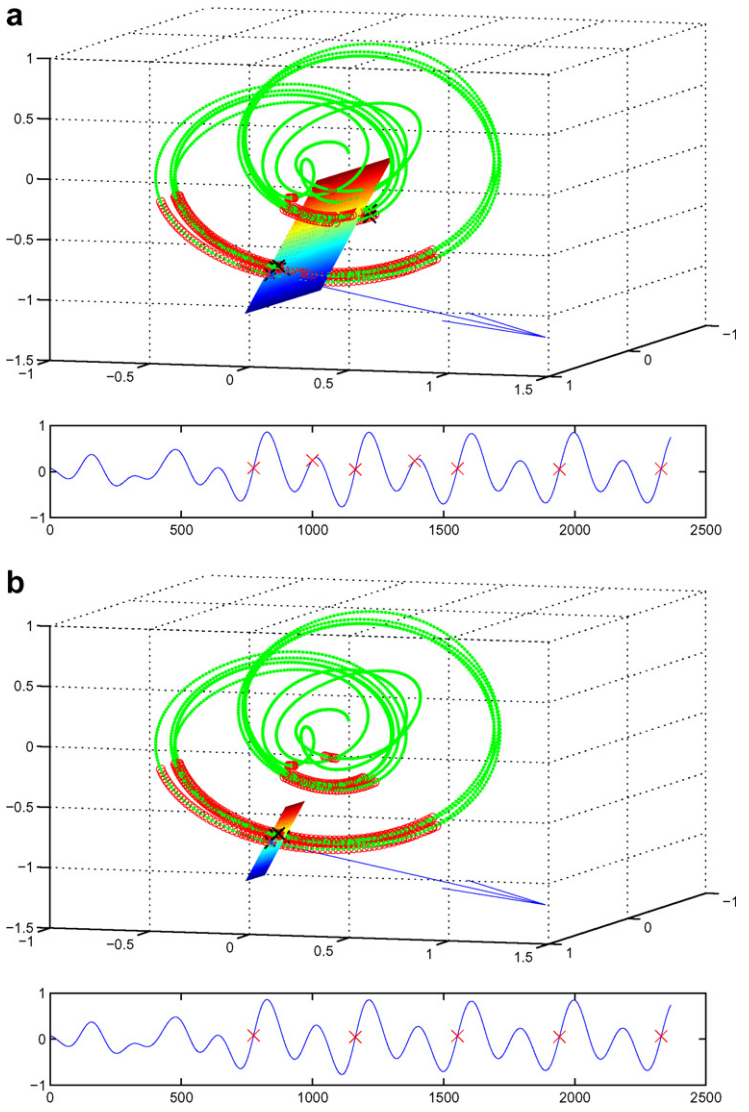


Fig. 7. Vowel 'a' from a dysphonic speaker. Top: Projection of state-space embedding in three-dimensions and Poincaré plane. Circles are neighbors. Bottom: Waveform plot. (a) Wrong parts of the trajectories chosen. (b) Correct placement of Poincaré plane and pitch marks.

$$(\boldsymbol{P} - \boldsymbol{x}(n_0)) \cdot \boldsymbol{f}(n_0) = 0, \tag{10}$$

where $\boldsymbol{P}$ is any point on the Poincaré plane.

To calculate the intersections with the plane, the points before and after the passing of the trajectory through the plane have to be found. If

$$\begin{aligned}&(\boldsymbol{x}(n+1) - \boldsymbol{x}(n_0)) \cdot \boldsymbol{f}(n_0) > 0, \text{ and}\\ &(\boldsymbol{x}(n) - \boldsymbol{x}(n_0)) \cdot \boldsymbol{f}(n_0) < 0.\end{aligned} \tag{11}$$

then the points $\boldsymbol{x}(n)$, $\boldsymbol{x}(n+1)$ are just before and after the plane or if

$$(\boldsymbol{x}(n) - \boldsymbol{x}(n_0)) \cdot \boldsymbol{f}(n_0) = 0, \tag{12}$$

then point $\boldsymbol{x}(n)$ lies exactly on the Poincaré plane.

The exact location of the intersection points is calculated by linear interpolation between the two points before and after the intersection $\boldsymbol{x}(n)$ and $\boldsymbol{x}(n+1)$. The points, which are at the intersection of the trajectory with the Poincaré plane, and their time indices are considered as pitch mark positions.

The length of one frame is chosen so that at least two periods of the expected minimum frequency fit into the frame. If the signal is quasi-periodic, the trajectory returns at least once into the chosen neighborhood and intersects the Poincaré hyperplane and a pitch mark can be detected. The hop size depends on the pitch mark in the current frame. The beginning of the following frame is set to the last pitch mark.

### 3.3. Post-processing

The voiced/unvoiced decision is based on different criteria. First, a frame is considered as unvoiced if the energy of the low-pass filtered signal is below a certain threshold. If the energy criterion does not detect an unvoiced frame it is further analyzed in the state space in case of high fluctuations of the fundamental periods in one frame by considering different parameters and calculating a cost for each candidate point. First, the Euclidean distance of the pitch mark candidate points to the query point is considered. Then the distance between all candidate points is considered and a grouping of the candi-

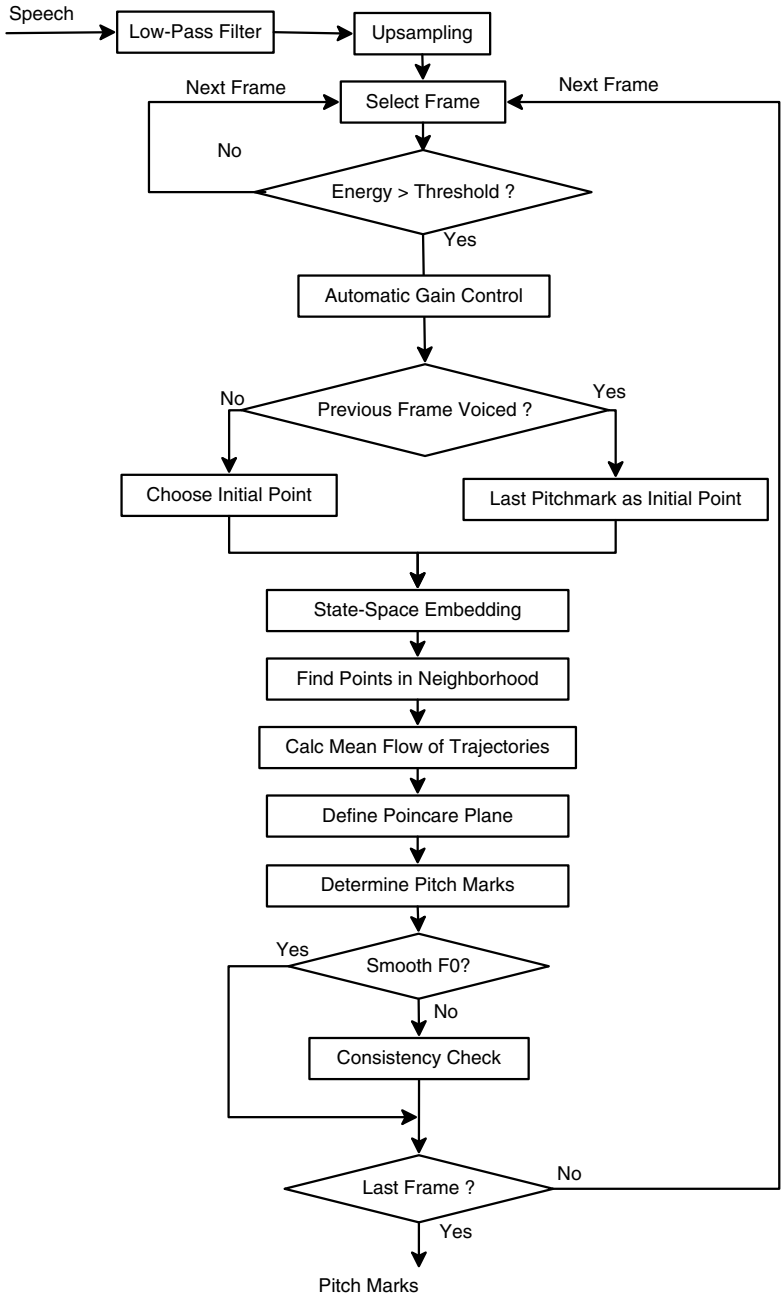


Fig. 8. Flow diagram of pitch marking system.

dates in two clusters is performed. If the result is a cluster which is clearly distinct from the points around the query point, those points are attributed a higher cost. Another parameter is the energy around a candidate pitch mark, a low energy is associated with a higher cost. Candidate points with a cost above a certain threshold are discarded. In addition to the detection of unvoiced sections, this also reduces the occasional hit of a first harmonic (Fig. 7).

Fig. 8 shows a flow diagram of the pitch marking system.

## 4. Discussion

There are some interesting results for the algorithm and open problems, which we want to discuss in this section.

### 4.1. Jitter in peak detection algorithm

Fig. 9 shows a signal section with a temporal evolution of its positive peaks. The positive signal part changes from a dual peak with its maximum on the right into a dual peak signal with its maximum on the left and back again. Peak picking pitch marking algorithms such as the peak picking algorithm from 'Praat' (see Appendix A.2.2) switch the pitch mark back and forth between the left and the right peak. This introduces a jitter, which is clearly an artefact of the algorithm. Since the Poincaré method is not directly based on time domain signal properties, the pitch cycles are followed correctly, staying at the same position over the whole signal section.

To have a quantitative result for comparison of the two methods, we calculate the local jitter, also called the period perturbation factor (PPF)

$$\mathrm{PPF} = \frac{100\%}{N-1}\sum_{i=1}^{N-1}\frac{u(i)-u(i-1)}{u(i)}, \tag{13}$$

where $u(i)$ is the period length sequence seen in Fig. 9. For the Poincaré method the result is PPF = 0.14% and for the Praat peak-picking method: PPF = 15.1%.

### 4.2. Diplophonic voice

In Fig. 10, a segment (vowel 'o') is taken out of a speech file. There, a short period of diplophonic fundamental frequency is present (sentence '*rl040—Judith found the manuscripts waiting for her on the piano*' from the Bagshaw database (Bagshaw, 1994)).

Other algorithms like Praat by Boersma and Weenink (2005) either fail for such events completely or detect a period doubling if the chosen minimum pitch value allows for such long pitch periods. The Poincaré method recognizes the rapidly alternating pitch periods correctly. Of course in this case it is a matter of definition whether the alternating period or the period doubling is the correct interpretation. Still, we consider our approach more useful, since the subharmonics can be derived from our result in a second step, if desired. This is not possible the other way round. If only the subharmonic is known, the period alteration in the time domain cannot be derived anymore.

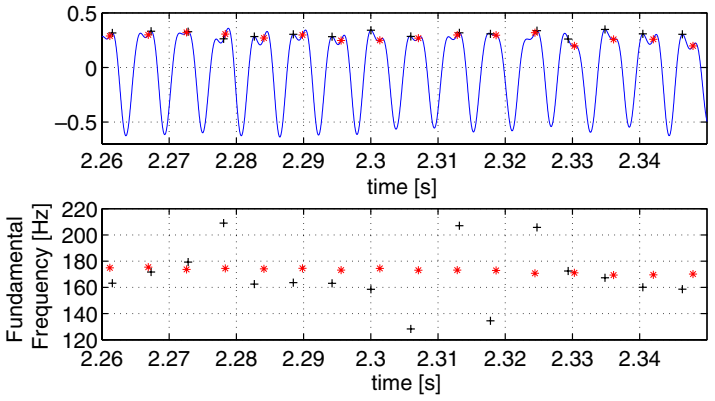


Fig. 9. Changing peaks. Comparison of results from the Poincaré method (∗) and the peak-picking method from 'Praat' (+). Top: Waveform with pitch marks. Bottom: Fundamental frequency estimates computed from pitch marks.

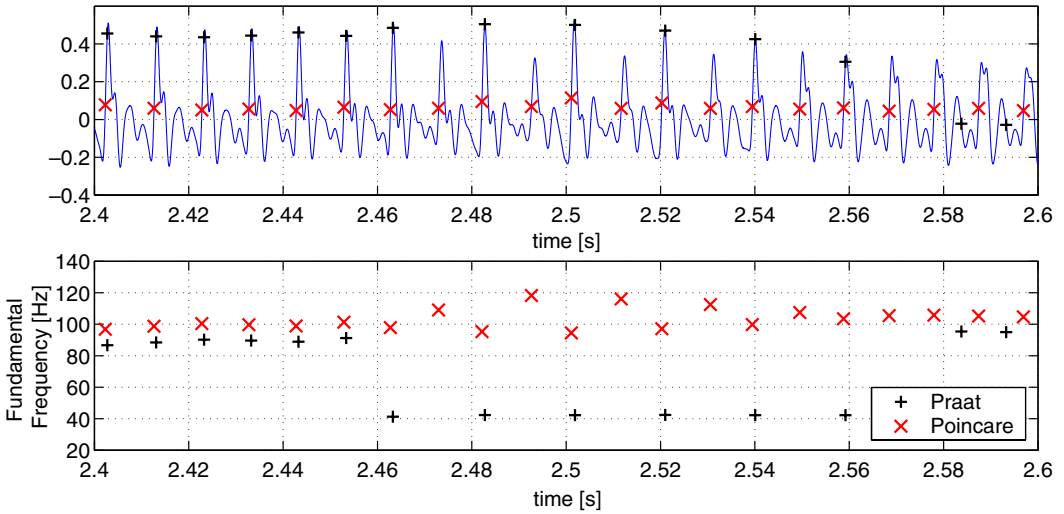


Fig. 10. Pitch marks of a diplophonic voice segment (vowel 'o'). Comparison of the results from the Poincaré method and 'Praat'.

In Fig. 11, the alternating size of the time-domain waveform peaks can be clearly seen in the state-space plot as two different bundles of the trajectory.

### 4.3. Phase drift

In Fig. 12, we see that the phase of the Poincaré pitch marks does not remain constant over the whole signal section seen in the plot. One can see that the marks slowly evolve from the positive to the negative peaks. This is a problem, which occurs only occasionally and a change of parameters usually removes the problem for one signal section, but might introduce a drift at another place.

However, no general design rule could be derived so far to completely remove this problem.

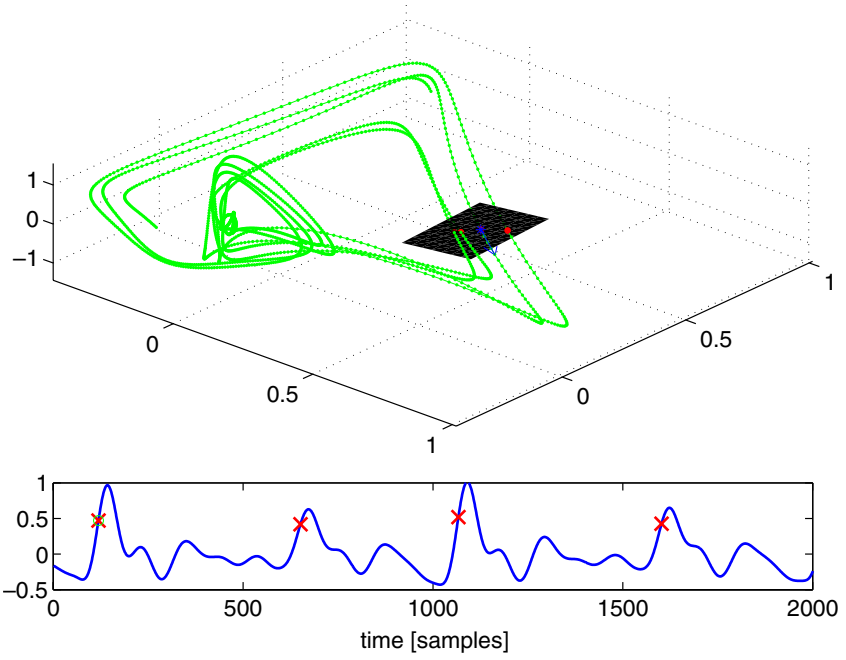


Fig. 11. State space representation of a segment of diplophonic voice. The two different states of the speech production system can be clearly seen.

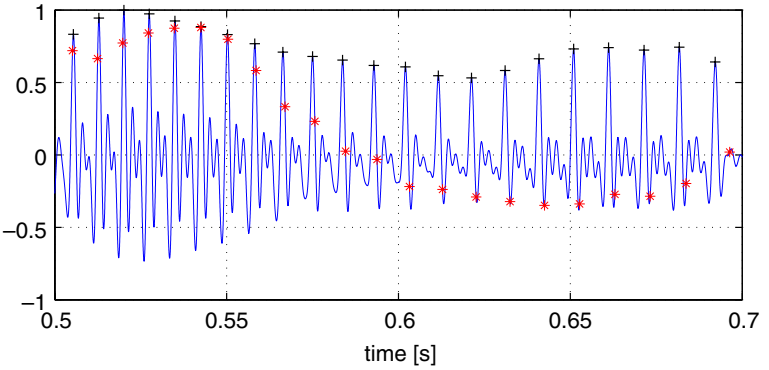


Fig. 12. Phasedrift of the pitch marks. The marks evolve from positive to negative peaks. Results from 'Praat' (+) and Poincaré (∗).

Specially for applications where high accuracy is an important issue or where the pitch mark should be synchronized with a given event, such as an energy maximum or a glottal closure instant, this may be a disadvantage of this method. The reason for this drift can either lie in the changing dynamics of the speech production system, or small errors in the calculation of the mean flow vector.

A phase drift was also reported by Mann (1999), but different to his approach we interpolate between two sampling points to determine the exact intersection of the trajectories with the Poincaré plane (see Section 3.2).

### 4.4. Resynchronization

In Fig. 13, the same signal section is shown twice after low-pass filtering with two different cut-off frequencies. After each unvoiced section, the initial point $\boldsymbol{x}(n_0)$ has to be set again. The search frame for the initial point is set to the first half of the current frame. As mentioned in Section 3.2, the position of the current frame depends on the last pitch mark of the previous frame (if voiced).

In the upper plot of Fig. 13, the initial point was set to the smaller peak at the beginning of the voiced section, consequently the following pitch marks are on the corresponding position in the following pitch cycles. The lower plot shows a setting, where the search frame was positioned to include the maximum peak of the pitch cycle, which is followed perfectly throughout the rest of the frame.

A possible solution would of course be to introduce a more sophisticated peak search algorithm, with some look-ahead, if a better choice for the initial point is available. This comes at the cost of reduced real-time capabilities of the algorithm, because the overall delay would be increased. If the initial points are chosen based on the bundling of the trajectories in state space, this is, of course,

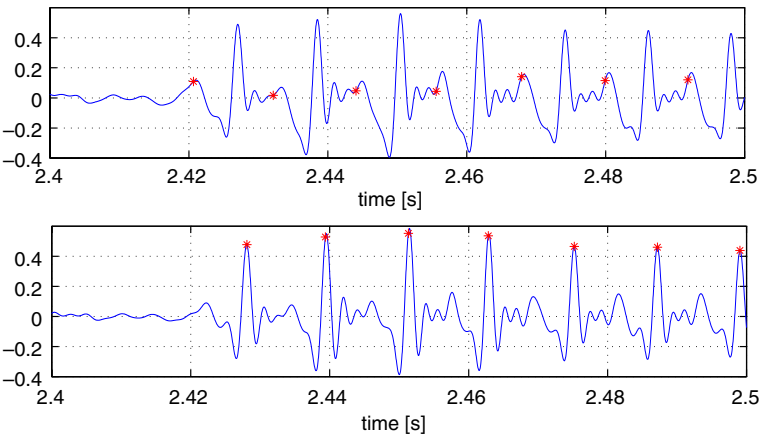


Fig. 13. Resynchronization after unvoiced period. Depending on the frame position different initial points are chosen.

not an issue since time-domain peaks are no criterion for this choice, anyway.

## 5. Evaluation

To evaluate the performance of the new pitch marking algorithm in a quantitative way, it is compared with a state-of-the-art pitch marking software. We choose the algorithm used by *Praat*, which is a software tool for speech processing and computer linguistics by Boersma and Weenink (2005). For pitch mark determination, *Praat* first does a sophisticated pitch determination using autocorrelation and a least-cost path search through the analyzed frames with several pitch candidates per frame. The resulting pitch contour provides a narrow search range for the pitch marking algorithm. It starts from an absolute extremum in a frame and determines the pitch marks by finding the cross-correlation maximum in the range given by the previously calculated pitch contour.

The database used for the evaluation of the pitch mark algorithm is a freely available pitch mark database. It includes speech and laryngeal signal samples of a male and a female subject and the corresponding pitch marks obtained by a pulse location algorithm applied to the laryngeal signal (Bagshaw, 1994).

### 5.1. Formal evaluation

Due to several issues discussed in Section 4, the evaluation of the algorithm cannot be performed assuming a fixed phase of the pitch marks. Since formal evaluation of the algorithm is important, an alternative method of evaluation is performed. The reference database contains a male and a female speaker with 50 short sentences each. For each sentence a file with reference pitch marks is available.

#### 5.1.1. Procedure

Since evaluation of pitch marks which are not associated with a fixed temporal event, such as a peak in the time domain, is difficult, a conversion to fundamental frequency ($F_0$) values for a fixed time step of 10 ms is performed. In case less than two pitch marks per frame are found the search window is extended to 20 ms, if still less than two pitch marks are found the frame is considered as unvoiced. For every frame an average value for the fundamental frequency is calculated. For the male files the range for the expected frequencies was set to 50–300 Hz, for the female files to 150–400 Hz. The procedure is rather an evaluation of fundamental frequency determination than the pitch marking capabilities of the algorithm. With this restriction, the performance of the algorithm can be compared to the results from 'Praat'.

Comparisons with other afore-mentioned related algorithms from the literature (Section 2.2) are difficult, since none of the referenced works deal with running speech, using full sentences (Kubin, 1997; Mann and McLaughlin, 1998). In addition they do not provide any formal evaluation results themselves.

### 5.2. Results

In Tables 1 and 2, the results for the chosen database are shown. For comparison, both the results for the Poincaré method and the 'Praat' pitch marking algorithm (Boersma and Weenink, 2005) are presented. In the upper part the voiced/unvoiced

Table 1
Results for a female speaker, 147 s of speech, 46.7% voiced speech

| | Voiced (%) | Voiced errors (%) | Unvoiced errors (%) |
|---|---|---|---|
| Poincaré | 50.3 | 3.0 | 2.2 |
| Praat | 51.6 | 4.4 | 0.6 |
| | Errors < 1% (%) | Errors < 5% (%) | Errors < 10% (%) |
| Poincaré | 71.7 | 91.7 | 95.8 |
| Praat | 70.7 | 91.2 | 95.7 |

The voiced column shows the percentage of detected voiced frames. The voiced errors are the percentage of frames falsely considered as voiced, and vice versa for the unvoiced errors. Errors < 1%, 5% and 10% show the percentage of pitch errors which are below 1%, 5% and 10% relative error, respectively.

Table 2
Results for a male speaker, 117 s of speech, 51.82% voiced speech

| | Voiced (%) | Voiced errors (%) | Unvoiced errors (%) |
|---|---|---|---|
| Poincaré | 54.6 | 2.7 | 8.7 |
| Praat | 57.7 | 5.8 | 4.8 |
| | Errors < 1% (%) | Errors < 5% (%) | Errors < 10% (%) |
| Poincaré | 74.8 | 96.1 | 98.3 |
| Praat | 69.1 | 93.2 | 97.6 |

The voiced column shows the percentage of detected voiced frames. The voiced errors are the percentage of frames falsely considered as voiced, and vice versa for the unvoiced errors. Errors < 1%, 5% and 10% show the percentage of pitch errors which are below 1%, 5% and 10% relative error, respectively.

errors are shown. In the first column the total ratio of voiced frames for the speech material can be seen, then the voiced errors and finally the unvoiced errors are shown. A voiced error is set if an unvoiced frame is falsely considered as voiced by the pitch detection algorithm. An unvoiced error is set if a voiced frame is falsely considered as unvoiced. Smaller numbers are better in this case. Voiced speech covers roughly half of the whole signal length, which is a reasonable rate to be able to evaluate voiced and unvoiced errors.

In the lower part of the table the accuracy of the results for the voiced frames is shown. Fundamental frequency errors are differentiated into the percentage of errors <1%, <5% and <10% of the reference fundamental frequency. It is desired to have a high percentage of the errors in the 'errors <1%' column.

*5.2.1. Discussion*

We see that the performance of the two algorithms is comparable. While the Poincaré method has slightly better results for the voiced error, Praat has a much lower error rate for unvoiced errors. Both algorithms can be tuned to favor either voiced or unvoiced errors, depending on the application. On the one hand, it can be better to be sure to capture all pitch marks while running the risk of having too many pitch marks. On the other hand, it can also be desirable to ensure that the pitch marks, which are found, are all correct ones while running the risk of not finding all pitch marks. In general, a balance between voiced and unvoiced errors at a low level is desired.

The accuracy of the voiced frames found is also very similar. For both algorithms roughly 70% of the $F_0$ values are below an error of 1%, while more than 90% of all errors are below the 5% error rate. For female speech, the results of the two algorithms are very close while for male speech the Poincaré method outperforms Praat by about 5% in its relative accuracy.

## 6. Conclusion

We presented an alternative approach to pitch mark determination, which applies methods from dynamical systems analysis to speech signal processing.

While the results are promising at the moment the new algorithm cannot outperform state-of-the-art pitch detection algorithms such as those used by 'Praat' in all situations. A clear advantage of the new algorithm has been shown for diplophonic voices. The rapidly alternating pitch periods are recognized correctly, where in contrast 'Praat' only detects the subharmonics. It has also been demonstrated, that peak-picking algorithms can introduce a jitter artefact if the analyzed signal has dual peaks, which vary in amplitude. In those cases, the Poincaré method follows the pitch cycles correctly.

One has to keep in mind, though, that the performance achieved by 'Praat' is much improved by the sophisticated pre- and post-processing algorithm, which considers previous and following frames next to the current one so as to improve the results. This higher-layer processing is not used by the presented new method, since it has been designed for a real-time application, and it was intended to keep the delay strictly of the order of a single frame.

## Appendix A

*A.1. Pseudo-code*

·Input speech signal $x(n)$
·Low-pass filter
·Upsample (if necessary)
·WHILE
index < lastindex − framelength
- Get segment with framelength at index
- IF energy(segment) < threshold,
  ·set frame unvoiced
  ·take next frame
- END
- Apply automatic gain control
- Normalize segment
- Choose initial point, $x(n_0)$, in time-domain
- Embed segment in pseudo-state space (dimension $M$, delay $\tau_d$)
  $\boldsymbol{x}(n) = [x(n), x(n-\tau_d), \ldots, x(n-(M-1)\tau_d)]$
- Select k neighbors in state space neighborhood $\mathcal{N}(n_0)$ of $\boldsymbol{x}(n_0)$
- Compute estimate of average vector flow $\boldsymbol{f}(n_0)$
  $\boldsymbol{f}(n_0) = \text{mean}(\boldsymbol{x}(n+1) - \boldsymbol{x}(n)) \quad \forall n \in \mathcal{N}(n_0)$
- Define Poincaré hyperplane perpendicular to $\boldsymbol{f}(n_0)$ going through $\boldsymbol{x}(n_0)(\boldsymbol{x}(n_0) - \boldsymbol{P}) \cdot \boldsymbol{f}(n_0) = 0$
- Calculate intersection of trajectories through Poincaré plane by interpolation between samples neighboring Poincaré plane

- IF std(T0) > 0.15 · median(T0)
  ·Discard points far away from $\boldsymbol{x}(n_0)$
- END
- Set index to beginning of next frame

·END WHILE
·Output pitch marks

### A.2. Praat

This is a short description of the pitch mark determination methods as implemented by 'Praat'. The description is copied from the Praat manual, which is also available online (see Boersma and Weenink (2005)).

As a first step, Praat runs a pitch determination algorithm to get a narrow search range for the pitch marks. It then can determine the pitch marks either with an algorithm based on cross correlation or an algorithm based on peak picking. The voiced intervals are determined on the basis of the voiced/unvoiced decisions in the pitch determination algorithm. For every voiced interval, a number of points (or glottal pulses) is found as follows.

#### A.2.1. Cross-correlation algorithm

(1) The first point $t_1$ is the absolute extremum of the amplitude of the sound, between $t_{\text{mid}} - \frac{T_0}{2}$ and $t_{\text{mid}} + \frac{T0}{2}$, where $t_{\text{mid}}$ is the midpoint of the interval, and $T_0$ is the period at $t_{\text{mid}}$, as can be interpolated from the pitch contour.
(2) From this point, we recursively search for points $t_i$ to the left until we reach the left edge of the interval. These points must be located between $t_{i-1} - 1.2 \cdot T_0(t_{i-1})$ and $t_{i-1} - 0.8 \cdot T_0(t_i - 1)$, and the cross-correlation of the amplitude in its environment $[t_i - T_0(t_i)/2; t_i + T_0(t_i)/2]$ with the amplitude of the environment of the existing point $t_{i-1}$ must be maximal (we use parabolic interpolation between samples of the correlation function).
(3) The same is done to the right of $t_1$.
(4) Though the voiced/unvoiced decision is initially taken by the Pitch contour, points are removed if their correlation value is less than 0.3; furthermore, one extra point may be added at the edge of the voiced interval if its correlation value is greater than 0.7.

#### A.2.2. Peak-picking algorithm

(1) The first point $t_1$ is the absolute extremum (or the maximum, or the minimum, depending the settings) of the amplitude of the sound, between $t_{\text{mid}} - \frac{T_0}{2}$ and $t_{\text{mid}} + \frac{T0}{2}$, where $t_{\text{mid}}$ is the midpoint of the interval, and $T_0$ is the period at $t_{\text{mid}}$, as can be interpolated from the pitch contour.
(2) From this point, we recursively search for points $t_i$ to the left until we reach the left edge of the interval. These points are the absolute extrema (or the maxima, or the minima) between the times $t_{i-1} - 1.2 \cdot T_0(t_{i-1})$ and $t_{i-1} - 0.8 \cdot T_0(t_{i-1})$.
(3) The same is done to the right of $t_1$.

## References


Bagshaw, P., 1994. Evaluating pitch determination algorithms, a pitch database. Available from: <http://www.cstr.ed.ac.uk/research/projects/fda/>.

Banbrook, M., Ushaw, G., McLaughlin, S., 1996. Lyapunov exponents from a time series: a noise-robust extraction algorithm. Chaos, Solitons Fractals 7 (7), 973–976.

Bimbot, F. (Ed.), 2003. ISCA Tutorial and Research Workshop on Non-linear Speech Processing (NOLISP). Le Croisic, France.

Boersma, P., Weenink, D., 2005. Praat, software for speech analysis and synthesis. Available from: <http://www.praat.org>, 3/2005.

Broomhead, D., King, G., 1986. On the qualitative analysis of experimental dynamical systems. In: Sarkar, S. (Ed.), Nonlinear Phenomena and Chaos. Adam Hilger, Bristol, UK, pp. 113–144.

Chollet, G., Esposito, A., Faundez-Zanuy, M., Marinaro, M. (Eds.), 2005. Nonlinear speech modeling and applications—advanced lectures and revised selected papers. Lecture Notes in Computer Science, 3445. Springer-Verlag GmbH.

Faundez-Zanuy, M., Janer, L., Esposito, A., Satue-Villar, A., Roure, J., Espinosa-Duro, V. (Eds.), 2006. Nonlinear Analyses and Algorithms for Speech Processing: International Conference on Non-Linear Speech Processing, NOLISP 2005. Lecture Notes in Computer Science, 3817. Springer-Verlag GmbH.

Giovanni, A., Ouaknine, M., Guelfucci, B., Yu, P., Zanaret, M., Triglia, J.-M., 1999a. Nonlinear behavior of vocal fold vibration: the role of coupling between the vocal folds. J. Voice 13 (4), 465–476.

Giovanni, A., Ouaknine, M., Triglia, J.-M., 1999b. Determination of largest Lyapunov exponents of vocal signals: application to unilateral paralysis. J. Voice 13 (3), 341–354.

Hagmüller, M., Kubin, G., 2003. Poincaré sections for pitch mark determination in dysphonic speech. In: Proc. 3rd Internat. Workshop on Models and Anal. of Vocal Emissions for Biomedical Appl. (MAVEBA). Firenze, Italy, December 10–12, pp. 281–284.

Hagmüller, M., Kubin, G., 2004. Voice enhancement of male speakers with laryngeal neoplasm. In: Proc. Internat. Conf. Spoken Language Process, Jeju Island, South Korea, October 4–8, pp. 541–544.

Hagmüller, M., Kubin, G., 2005. Poincaré sections for pitch mark determination. In: ISCA Tutorial and Research Workshop (ITRW) on Non-Linear Speech Process., Barcelona, Spain, April 19–22, pp. 107–113.

Hegger, R., Kantz, H., Matassini, L., 2000. Denoising human speech signals using chaos-like features. Phys. Rev. Lett. 84, 3197.

Hegger, R., Kantz, H., Matassini, L., 2001. Noise reduction for human speech signals by pdf projections in embedding spaces. IEEE Trans. Circuits Systems I: Fundamental Theory Appl. 48 (12), 1454–1461.

Herzel, H., Berry, D., Titze, I., Steinecke, I., 1995. Nonlinear dynamics of the voice: signal analysis and biomechanical modeling. Chaos 5 (1), 30–34.

Indrebo, K.M., Povinelli, R.J., Johnson, M.T., 2006. Sub-banded reconstructed phase spaces for speech recognition. Speech Commun. 48 (7), 760–774.

Jiang, J.J., Zhang, Y., McGilligan, C., 2006. Chaos in voice, from modeling to measurement. J. Voice 20 (1), 2–17.

Johnson, M., Lindgren, A., Povinelli, R., Yuan, X., 2003. Performance of nonlinear speech enhancement using phase space reconstruction. In: Proc. IEEE Internat. Conf. on Acoustics, Speech, and Signal Process., Vol. 1. Hong Kong, April 6–10, pp. 920–923.

Kantz, H., Schreiber, T., 2004. Nonlinear Time Series Analysis, 2nd ed. Cambridge University Press.

Kleijn, W.B., 2002. Enhancement of coded speech by constrained optimization. In: Proc. IEEE Workshop on Speech Coding. Tsukuba, Ibaraki, Japan.

Kokkinos, I., Maragos, P., 2005. Nonlinear speech analysis using models for chaotic systems. IEEE Trans. Speech Audio Process. 13 (6), 1098–1109.

Kubin, G., 1995. Nonlinear processing of speech. In: Kleijn, W., Paliwal, K.K. (Eds.), Speech Coding and Synthesis. Elsevier, pp. 557–610 (Chapter 16).

Kubin, G. 1997. Poincaré section techniques for speech. In: Proc. IEEE Workshop on Speech Coding for Telecommun. '97. Pocono Manor, PA, pp. 7–8.

Kumar, A., Mullick, S.K., 1996. Nonlinear dynamical analysis of speech. J. Acoust. Soc. Amer. 100 (1), 615–629.

Little, M., McSharry, P., Moroz, I., Roberts, S., 2006. Testing the assumptions of linear prediction analysis in normal vowels. J. Acoust. Soc. Amer. 119 (1), 549–558.

Mann, I., 1999. An investigation of nonlinear speech synthesis and pitch modification techniques. Ph.D. Thesis, University of Edinburgh.

Mann, I., McLaughlin, S., 1998. A nonlinear algorithm for epoch marking in speech signals using Poincaré maps. In: Proc. Eur. Signal Process. Conf., Vol. 2. Rhodes, Greece, September, pp. 701–704.

Matassini, L., Manfredi, C., 2002. Software correction of vocal disorders. Comput. Methods Programs Biomed. 68 (2), 135–145.

Moulines, E., Laroche, J., 1995. Non-parametric techniques for pitch-scale and time-scale modification of speech. Speech Commun. 16, 175–205.

Nöth, E., Batliner, A., Warnke, V., Haas, J., Boros, M., Buckow, J., Huber, R., Gallwitz, F., Nutt, M., Niemann, H., 2002. On the use of prosody in automatic dialogue understanding. Speech Commun. 36 (1–2), 45–62.

Sauer, T., Yorke, J.A., Casdagli, M., 1991. Embedology. J. Stat. Phys. 65 (3/4), 579–616.

Schoentgen, J., 2003. Decomposition of vocal cycle length perturbations into vocal jitter and vocal microtremor, and comparison of their size in normophonic speakers. J. Voice 17 (2), 114–125.

Schreiber, T., 1995. Efficient neighbor searching in nonlinear time series analysis. Int. J. Bifurcat. Chaos 5, 349–358.

Stylianou, Y., Laroche, J., Moulines, E., 1995. High-quality speech modification based on a harmonic + noise model. In: Proc. Eur. Conf. on Speech Commun. and Technol., Madrid, Spain, September 18–21, pp. 451–454.

Takens, F., 1981. Detecting strange attractors in turbulence Lecture Notes in Mathematics, 898. Springer, New York, pp. 366–381.

Terez, D., 2002. Robust pitch determination using nonlinear state-space embedding. In: Proc. IEEE Internat. Conf. on Acoustics, Speech, and Signal Process., Vol. 1. Orlando, Florida, pp. 345–348.

Tishby, N., 1990. A dynamical systems approach to speech processing. In: Proc. IEEE Internat. Conf. on Acoustics, Speech, and Signal Process., Vol. 4. Albuquerque, NM, April, pp. 365–368.

Titze, I.R., 1994. Workshop on acoustic voice analysis—summary statement. In: Proc. Workshop on Acoustic Voice Analysis. Denver, Colorado, February.