EURASIP Journal on Applied Signal Processing 2003:8, 791–805

Parameter Estimation of a Plucked String Synthesis Model Using a Genetic Algorithm with Perceptual Fitness Calculation

Janne Riionheimo

Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology, P.O. Box 3000, FIN-02015 HUT, Espoo, Finland
Email: janne.riionheimo@hut.fi

Vesa Välimäki

Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology, P.O. Box 3000, FIN-02015 HUT, Espoo, Finland

Pori School of Technology and Economics, Tampere University of Technology, P.O. Box 300, FIN-28101, Pori, Finland
Email: vesa.valimaki@hut.fi

Received 30 June 2002 and in revised form 2 December 2002

We describe a technique for estimating control parameters for a plucked string synthesis model using a genetic algorithm. The model has been intensively used for sound synthesis of various string instruments but the fine tuning of the parameters has been carried out with a semiautomatic method that requires some hand adjustment with human listening. An automated method for extracting the parameters from recorded tones is described in this paper. The calculation of the fitness function utilizes knowledge of the properties of human hearing.

Keywords and phrases: sound synthesis, physical modeling synthesis, plucked string synthesis, parameter estimation, genetic algorithm.

1. INTRODUCTION

Model-based sound synthesis is a powerful tool for creating natural sounding tones by simulating the sound production mechanisms and physical behavior of real musical instruments. These mechanisms are often too complex to simulate in every detail, so simplified models are used for synthesis. The aim is to generate a perceptually indistinguishable model for real instruments.

One workable method for physical modelling synthesis is based on digital waveguide theory proposed by Smith [1]. In the case of the plucked string instruments, the method can be extended to model also the plucking style and instrument body [2, 3]. A synthesis model of this kind can be applied to synthesize various plucked string instruments by changing the control parameters and using different body and plucking models [4, 5]. A characteristic feature in string instrument tones is the double decay and beating effect [6], which can be implemented by using two slightly mistuned string models in parallel to simulate the two polarizations of the transversal vibratory motion of a real string [7].

Parameter estimation is an important and difficult challenge in sound synthesis. Usually, the natural parameter settings are in great demand at the initial state of the synthesis. When using these parameters with a model, we are able to produce real-sounding instrument tones. Various methods for adjusting the parameters to produce the desired sounds have been proposed in the literature [4, 8, 9, 10, 11, 12]. An automated parameter calibration method for a plucked string synthesis model has been proposed in [4, 8], and then improved in [9]. It gives the estimates for the fundamental frequency, the decay parameters, and the excitation signal which is used in commuted synthesis.

Our interest in this paper is the parameter estimation of the model proposed by Karjalainen et al. [7]. The parameters of the model have earlier been calibrated automatically, but the fine-tuning has required some hand adjustment. In this work, we use recorded tones as a target sound with which the synthesized tones are compared. All synthesized sounds are then ranked according to their similarity with the recorded tone. An accurate way to measure sound quality from the

viewpoint of auditory perception would be to carry out listening tests with trained participants and rank the candidate solutions according to the data obtained from the tests [13]. This method is extremely time consuming and, therefore, we are forced to use analytical methods to calculate the quality of the solutions. Various techniques to simulate human hearing and calculate perceptual quality exist. Perceptual linear predictive (PLP) technique is widely used with speech signals [14], and frequency-warped digital signal processing is used to implement perceptually relevant audio applications [15].

In this work, we use an error function that simulates the human hearing and calculates the perceptual error between the tones. Frequency masking behavior, frequency dependence, and other limitations of human hearing are taken into account. From the optimization point of view, the task is to find the global minimum of the error function. The variables of the function, that is, the parameters of the synthesis model, span the parameter space where each point corresponds to a set of parameters and thus to a synthesized sound. When dealing with discrete parameter values, the number of parameter sets is finite and given by the product of the number of possible values of each parameter. Using nine control parameters with 100 possible values, a total of 10^{18} combinations exist in the space and, therefore, an exhaustive search is obviously impossible.

Evolutionary algorithms have shown a good performance in optimizing problems relating to the parameter estimation of synthesis models. Vuori and Välimäki [16] tried a simulated evolution algorithm for the flute model, and Horner et al. [17] proposed an automated system for parameter estimation of FM synthesizer using a genetic algorithm (GA). GAs have been used for automatically designing sound synthesis algorithms in [18, 19]. In this study, a GA is used to optimize the perceptual error function.

This paper is sectioned as follows. The plucked string synthesis model and the control parameters to be estimated are described in Section 2. Parameter estimation problem and methods for solving it are discussed in Section 3. Section 4 concentrates on the calculation of the perceptual error. In Section 5, we discretize the parameter space in a perceptually reasonable manner. Implementation of the GA and different schemes for selection, mutation, and crossover used in our work are surveyed in Section 6. Experiments and results are analyzed in Section 7 and conclusions are finally drawn in Section 8.

2. PLUCKED STRING SYNTHESIS MODEL

The model proposed by Karjalainen et al. [7] is used for plucked string synthesis in this study. The block diagram of the model is presented in Figure 1. It is based on digital waveguide synthesis theory [1] that is extended in accordance with commuted waveguide synthesis approach [2, 3] to include also the body modes of the instrument in the string synthesis model.

Different plucking styles and body responses are stored as wavetables in the memory and used to excite the two string

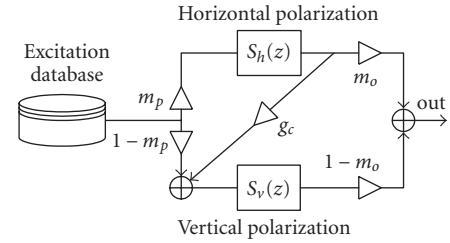


Figure 1: The plucked string synthesis model.

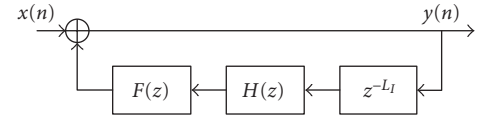


Figure 2: The basic string model.

models $S_h(z)$ and $S_v(z)$ that simulate the effect of the two polarizations of the transversal vibratory motion. A single string model $S(z)$ in Figure 2 consists of a lowpass filter $H(z)$ that controls the decay rate of the harmonics, a delay line z^{-L_I}, and a fractional delay filter $F(z)$. The delay time around the loop for a given fundamental frequency f_0 is

$$L_d = \frac{f_s}{f_0}, \tag{1}$$

where f_s is the sampling rate (in Hz). The loop delay L_d is implemented by the delay line z^{-L_I} and the fractional delay filter $F(z)$. The delay line is used to control the integer part L_I of the string length while the coefficients of the filter $F(z)$ are adjusted to produce the fractional part L_f [20]. The fractional delay filter $F(z)$ is implemented as a first-order allpass filter. Two string models are typically slightly mistuned to produce a natural sounding beating effect.

A one-pole filter with transfer function

$$H(z) = g\frac{1+a}{1+az^{-1}} \tag{2}$$

is used as a loop filter in the model. Parameter $0 < g < 1$ in (2) determines the overall decay rate of the sound while parameter $-1 < a < 0$ controls the frequency-dependent decay. The excitation signal is scaled by the mixing coefficients m_p and $(1 - m_p)$ before sending it to two string models. Coefficient g_c enables coupling between the two polarizations. Mixing coefficient m_o defines the proportion of the two polarizations in the output sound. All parameters m_p, g_c, and m_o are chosen to have values between 0 and 1. The transfer function of the entire model is written as

$$\begin{aligned} M(z) = {} & m_p m_o S_h(z) + (1 - m_p)(1 - m_o)S_v(z) \\ & + m_p(1 - m_o)g_c S_h(z)S_v(z), \end{aligned} \tag{3}$$

Table 1: Control parameters of the synthesis model.

Parameter	Control
$f_{0,h}$	Fundamental frequency of the horizontal string model
$f_{0,v}$	Fundamental frequency of the vertical string model
g_h	Loop gain of the horizontal string model
a_h	Frequency-dependent gain of the horizontal string model
g_v	Loop gain of the vertical string model
a_v	Frequency-dependent gain of the vertical string model
m_p	Input mixing coefficient
m_o	Output mixing coefficient
g_c	Coupling gain of the two polarizations

where the string models $S_h(z)$ and $S_v(z)$ for the two polarizations can be written as an individual string model

$$S(z) = \frac{1}{1 - z^{-L_I} F(z) H(z)}. \quad (4)$$

Synthesis model of this kind has been intensively used for sound synthesis of various plucked string instruments [5, 21, 22]. Different methods for estimating the parameters have been used, but in consequence of interaction between the parameters, systematic methods are at least troublesome but probably impossible. The nine parameters that are used to control the synthesis model are listed in Table 1.

3. ESTIMATION OF THE MODEL PARAMETERS

Determination of the proper parameter values for sound synthesis systems is an important problem and also depends on the purpose of the synthesis. When the goal is to imitate the sounds of real instruments, the aim of the estimation is unambiguous: we wish to find a parameter set which gives the sound output that is sufficiently similar to the natural one in terms of human perception. These parameters are also feasible for virtual instruments at the initial stage after which the limits of real instruments can be exceeded by adjusting the parameters in more creative ways.

Parameters of a synthesis model correspond normally to the physical characteristics of an instrument [7]. The estimation procedure can then be seen as sound analysis where the parameters are extracted from the sound or from the measurements of physical behavior of an instrument [23]. Usually, the model parameters have to be fine-tuned by laborious trial and error experiments, in collaboration with accomplished players [23]. Parameters for the synthesis model in Figure 1 have earlier been estimated this way and recently in a semiautomatic fashion, where some parameter values can be obtained with an estimation algorithm while others must be guessed. Another approach is to consider the parameter estimation problem as a nonlinear optimization process and take advantage of the general searching methods. All possible parameter sets can then be ranked according to their similarity with the desired sound.

3.1. Calibrator

A brief overview of the calibration scheme, used earlier with the model, is given here. The fundamental frequency $\hat{f}_0$ is first estimated using the autocorrelation method. The frequency estimate in samples from (1) is used to adjust the delay line length L_I and the coefficients of the fractional delay filter $F(z)$. The amplitude, frequency, and phase trajectories for partials are analyzed using the short-time Fourier transform (STFT), as in [4]. The estimates for loop filter parameters g and a are then analyzed from the envelopes of individual partials. The excitation signal for the model is extracted from the recorded tone by a method described in [24]. The amplitude, frequency, and phase trajectories are first used to synthesize the deterministic part of the original signal and the residual is obtained by a time-domain subtraction. This produces a signal which lacks the energy to excite the harmonics when used with the synthesis model. This is avoided by inverse filtering the deterministic signal and the residual separately. The output signal of the model is finally fed to the optimization routine which automatically fine-tunes the model parameters by analyzing the time-domain envelope of the signal.

The difference in the length of the delay lines can be estimated based on the beating of a recorded tone. In [25], the beating frequency is extracted from the first harmonic of a recorded string instrument tone by fitting a sine wave using the least squares method. Another procedure for extracting beating and two-stage decay from the string tones is described by Bank in [26]. In practice, the automatical calibrator algorithm is first used to find decent values for the control parameters of one string model. These values are also used for another string model. The mistuning between the two string models has then been found by ear [5] and the differences in the decay parameters are set by trial and error. Our method automatically extracts the nine control parameter values from recorded tones.

3.2. Optimization

Instead of extracting the parameters from audio measurements, our approach here is to find the parameter set that produces a tone that is perceptually indistinguishable from the target one. Each parameter set can be assigned with a

quality value which denotes how good is the candidate solution. This performance metric is usually called a fitness function, or inversely, an error function. A parameter set is fed into the fitness function which calculates the error between the corresponding synthesized tone and the desired sound. The smaller the error, the better the parameter set and the higher the fitness value. These functions give a numerical grade to each solution, by means of which we are able to classify all possible parameter sets.

4. FITNESS CALCULATION

Human hearing analyzes sound both in the frequency and time domain. Since spectra of all musical sounds vary with time, it is appropriate to calculate the spectral similarity in short time segments. A common method is to measure the least squared error of the short-time spectra of the two sounds [17, 18]. The STFT of signal $y(n)$ is a sequence of discrete Fourier transforms (DFT)

$$Y(m,k) = \sum_{n=0}^{N-1} w(n)y(n+mH)e^{-jw_k n}, \quad m = 0, 1, 2, \ldots, \tag{5}$$

with

$$w_k = \frac{2\pi k}{N}, \quad k = 0, 1, 2, \ldots, N-1, \tag{6}$$

where N is the length of the DFT, $w(n)$ is a window function, and H is the hop size or time advance (in samples) per frame. Integers m and k refer to the frame index and frequency bin, respectively. When N is a power of two, for example, 1024, each DFT can be computed efficiently with the FFT algorithm. If $o(n)$ is the output sound of the synthesis model and $t(n)$ is the target sound, then the error (inverse of the fitness) of the candidate solution is calculated as follows:

$$E = \frac{1}{F} = \frac{1}{L}\sum_{m=0}^{L-1}\sum_{k=0}^{N-1}\left(|O(m,k)| - |T(m,k)|\right)^2, \tag{7}$$

where $O(m,k)$ and $T(m,k)$ are the STFT sequences of $o(n)$ and $t(n)$ and L is the length of the sequences.

4.1. Perceptual quality

The analytical error calculated from (7) is a raw simplification from the viewpoint of auditory perception. Therefore, an auditory model is required. One possibility would be to include the frequency masking properties of human hearing by applying a narrow band masking curve [27] for each partial. This method has been used to speed up additive synthesis [28] and perceptual wavetable matching for synthesis of musical instrument tones [29]. One disadvantage of the method is that it requires peak tracking of partials, which is a time-consuming procedure. We use here a technique which determines the threshold of masking from the STFT sequences. The frequency components below that threshold are inaudible, therefore, they are unnecessary when calculating the perceptual similarity. This technique proposed in [30] has been successfully applied in audio coding and perceptual error calculation [18].

4.2. Calculating the threshold of masking

The threshold of masking is calculated in several steps:

(1) windowing the signal and calculating STFT,
(2) calculating the power spectrum for each DFT,
(3) mapping the frequency scale into the Bark domain and calculating the energy per critical band,
(4) applying the spreading function to the critical band energy spectrum,
(5) calculating the spread masking threshold,
(6) calculating the tonality-dependent masking threshold,
(7) normalizing the raw masking threshold and calculating the absolute threshold of masking.

The frequency power spectrum is translated into the Bark scale by using the approximation [27]

$$\nu = 13\arctan\left(\frac{0.76f}{\text{kHz}}\right) + 3.5\arctan\left(\frac{f}{7.5\,\text{kHz}}\right)^2, \tag{8}$$

where f is the frequency in Hertz and ν is the mapped frequency in Bark units. The energy in each critical band is calculated by summing the frequency components in the critical band. The number of critical bands depends on the sampling rate and is 25 for the sample rate of 44.1 kHz. The discrete representation of fixed critical bands is a close approximation and, in reality, each band builds up around a narrow band excitation. A power spectrum $P(k)$ and energy per critical band $Z(\nu)$ for a 12 milliseconds excerpt from a guitar tone are shown in Figure 3a.

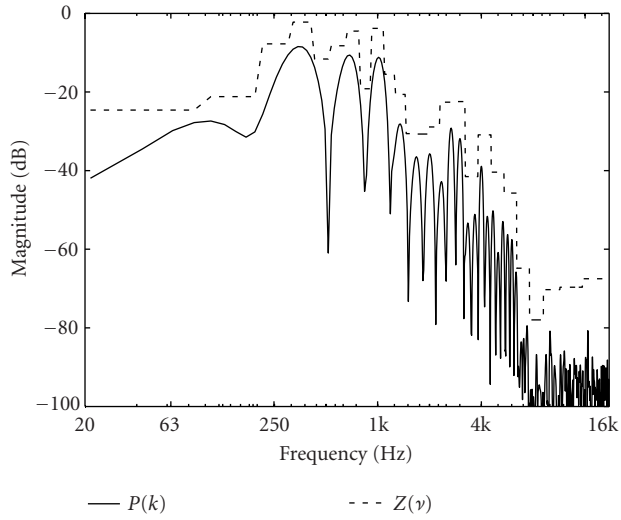
The effect of masking of each narrow band excitation spreads across all critical bands. This is described by a spreading function given in [31]

$$10\log_{10} B(\nu) = 15.91 + 7.5(\nu + 0.474) - 17.5\sqrt{1 + (\nu + 0.474)^2}\ \text{dB}. \tag{9}$$

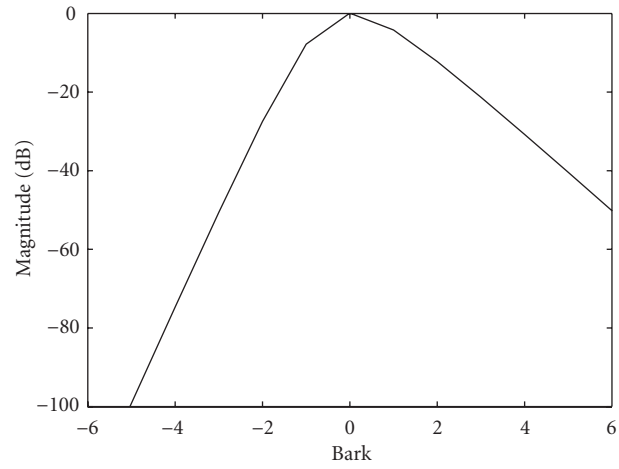
The spreading function is presented in Figure 3b. The spreading effect is applied by convolving the critical band energy function $Z(\nu)$ with the spreading function $B(\nu)$ [30]. The spread energy per critical band $S_P(\nu)$ is shown in Figure 3c.

The masking threshold depends on the characteristics of the masker and masked tone. Two different thresholds are detailed and used in [30]. For the tone masking noise, the threshold is estimated as $14.5 + \nu$ dB below the S_P. For noise masking, the tone it is estimated as 5.5 dB below the S_P. A spectral flatness measure is used to determine the noiselike or tonelike characteristics of the masker. The spectral flatness measure V is defined in [30] as the ratio of the geometric to the arithmetic mean of the power spectrum. The tonality factor α is defined as follows:

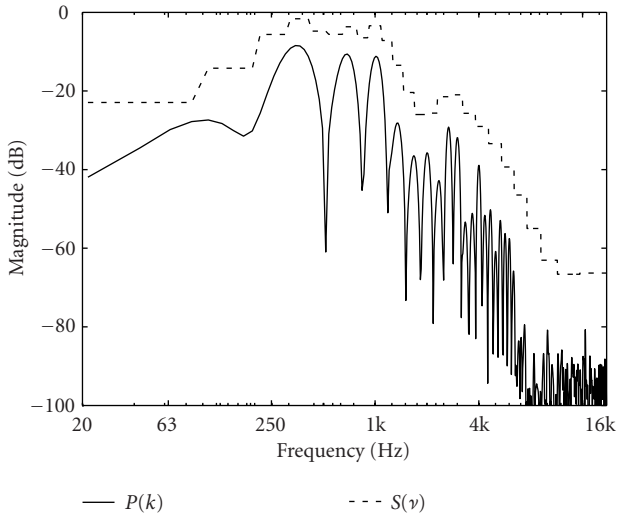
$$\alpha = \min\left(\frac{V}{V_{\max}}, 1\right), \tag{10}$$



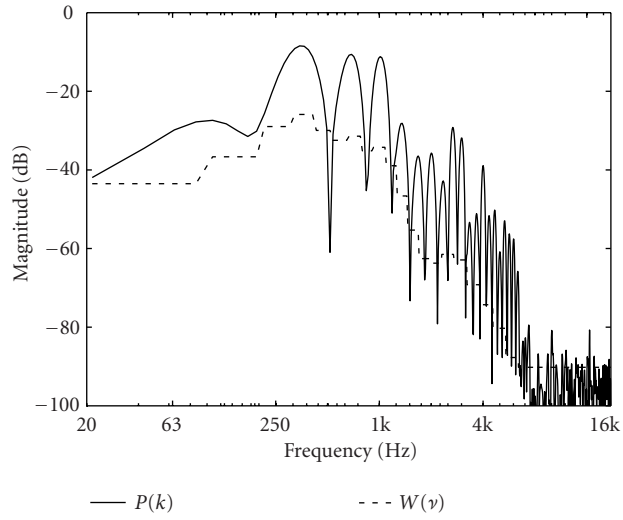
(a) Power spectrum (solid line) and energy per critical band (dashed line).



(b) Spreading function.



(c) Power spectrum (solid line) and spread energy per critical band (dashed line).



(d) Power spectrum (solid line) and final masking threshold (dashed line).

FIGURE 3: Determining the threshold of masking for a 12 milliseconds excerpt from a recorded guitar tone. Fundamental frequency of the tone is 331 Hz.

where $V_{\max} = -60$ dB. That is to say that if the masker signal is entirely tonelike, then $\alpha = 1$, and if the signal is pure noise, then $\alpha = 0$. The tonality factor is used to geometrically weight the two thresholds mentioned above to form the masking energy offset $U(\nu)$ for a critical band

$$U(\nu) = \alpha(14.5 + \nu) + 5.5(1 - \alpha). \tag{11}$$

The offset is then subtracted from the spread spectrum to estimate the raw masking threshold

$$R(\nu) = 10^{\log_{10}(S_P(\nu)) - U(\nu)/10}. \tag{12}$$

Convolution of the spreading function and the critical band energy function increases the energy level in each band. The normalization procedure used in [30] takes this into account and divides each component of $R(\nu)$ by the number of points in the corresponding band

$$Q(\nu) = \frac{R(\nu)}{N_p}, \tag{13}$$

where N_p is the number of points in the particular critical band. The final threshold of masking for a frequency spectrum $W(k)$ is calculated by comparing the normalized threshold to the absolute threshold of hearing and mapping from Bark to the frequency scale. The most sensitive area in human hearing is around 4 kHz. If the normalized

energy $Q(\nu)$ in any critical band is lower than the energy in a 4 kHz sinusoidal tone with one bit of dynamic range, it is changed to the absolute threshold of hearing. This is a simplified method to set the absolute levels since in reality the absolute threshold of hearing varies with the frequency.

An example of the final threshold of masking is shown in Figure 3d. It is seen that many of the high partials and the background noise at the high frequencies are below the threshold and thus inaudible.

4.3. Calculating the perceptual error

Perceptual error is calculated in [18] by weighting the error from (7) with two matrices

$$G(m,k) = \begin{cases} 1 & \text{if } T(m,k) \geq W(m,k), \\ 0 & \text{otherwise,} \end{cases}$$

$$H(m,k) = \begin{cases} 1 & \text{if } O(m,k) \geq W(m,k),\ T(m,k) < W(m,k), \\ 0 & \text{otherwise,} \end{cases} \tag{14}$$

where m and k refer to the frame index and frequency bin, as defined previously. Matrices are defined such that the full error is calculated for spectral components which are audible in a recorded tone $t(n)$ (that is above the threshold of masking). The matrix $G(m,k)$ is used to account for these components. For the components which are inaudible in a recorded tone but audible in the sound output of the model $o(n)$, the error between the sound output and the threshold of masking is calculated. The matrix $H(m,k)$ is used to weight these components.

Perceptual error E_p is a sum of these two cases. No error is calculated for the components which are below the threshold of masking in both sounds. Finally, the perceptual error function is evaluated as

$$E_p = \frac{1}{F_p} = \frac{1}{L}\sum_{k=0}^{N-1} W_s(k) \sum_{m=0}^{L-1} \left[\left(|O(m,k)| - |T(m,k)| \right)^2 G(m,k) \right] + \left[\left(|O(m,k)| - |T(m,k)| \right)^2 H(m,k) \right], \tag{15}$$

where $W_s(k)$ is an inverted equal loudness curve at sound pressure level of 60 dB shown in Figure 4 that is used to weight the error and imitate the frequency-dependent sensitivity of human hearing.

5. DISCRETIZING THE PARAMETER SPACE

The number of data points in the parameter space can be reduced by discretizing the individual parameters in a perceptually reasonable manner. The range of parameters can be reduced to cover only all the possible musical tones and deviation steps can be kept just below the discrimination threshold.

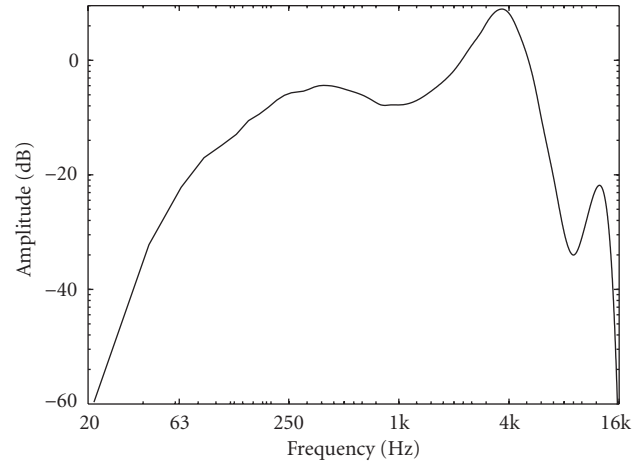


Figure 4: The frequency-dependent weighting function, which is the inverse of the equal loudness curve at the SPL of 60 dB.

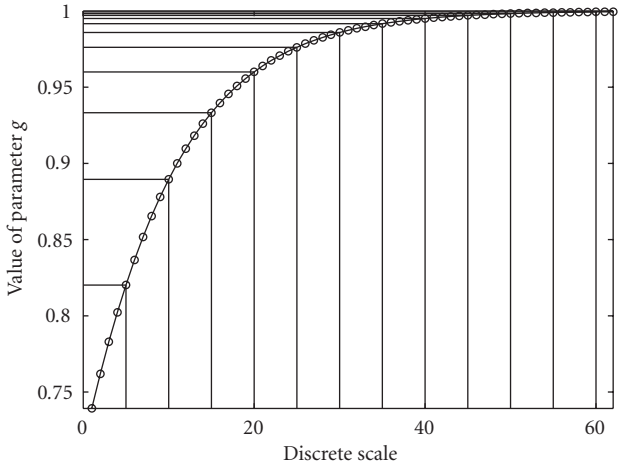
5.1. Decay parameters

The audibility of variations in decay of the single string model in Figure 2 have been studied in [32]. Time constant τ of the overall decay was used to describe the loop gain parameter g while the frequency-dependent decay was controlled directly by parameter a. Values of τ and a were varied and relatively large deviations in parameters were claimed to be inaudible. Järveläinen and Tolonen [32] proposed that a variation of the time constant between 75% and 140% of the reference value can be allowed in most cases. An inaudible variation for the parameter a was between 83% and 116% of the reference value.

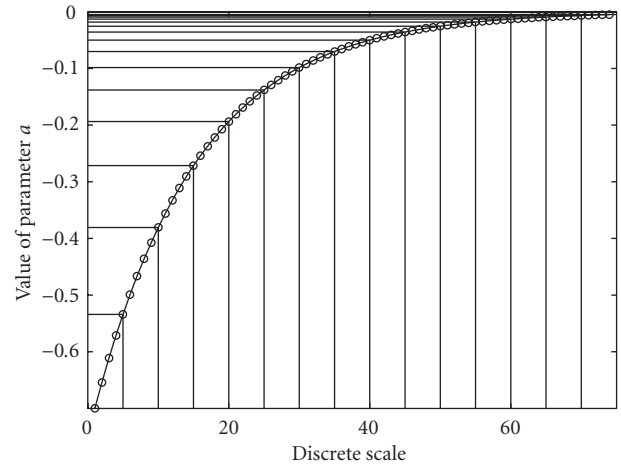
The discrimination thresholds were determined with two different tone durations 0.6 second and 2.0 seconds. In our study, the judgement of similarity between two tones is done by comparing the entire signals and, therefore, the results from [32] cannot be directly used for the parametrization of a and g. The tolerances are slightly smaller because the judgement is made based on not only the decay but also the duration of a tone. Based on our informal listening test and including a margin of certainty, we have defined the variation to be 10% for the τ and 7% for the parameter a. The parameters are bounded so that all the playable musical sounds from tightly damped picks to very slowly decaying notes are possible to produce with the model. This results in 62 discrete nonuniformly distributed values for g and 75 values for a, as shown in Figures 5a and 5b. The corresponding amplitude envelopes of tones with different g parameter are shown in Figure 5c. Loop filter magnitude responses for varying parameter a with $g = 1$ are shown in Figure 5d.

5.2. Fundamental frequency and beating parameters

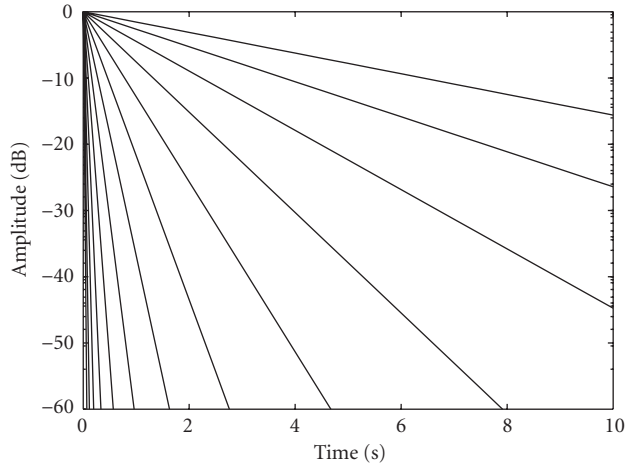
The fundamental frequency estimate $\hat{f}_0$ from the calibrator is used as an initial value for both polarizations. When the



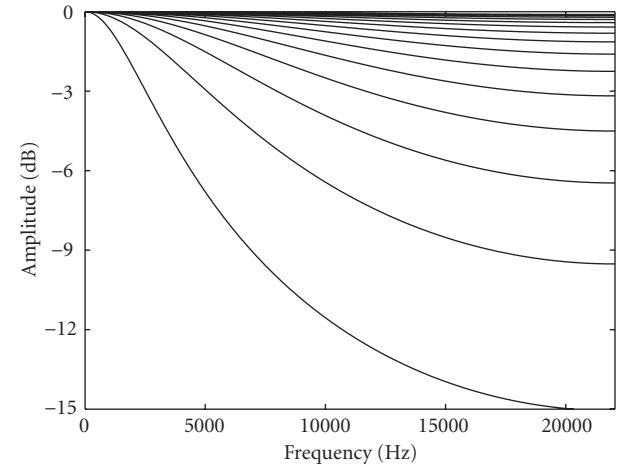
(a) Discrete values for the parameter g when $f_0 = 331$ and the variation for the time constant τ is 10%.



(b) Discrete values for the parameter a when the variation is 7%.



(c) Amplitude envelopes of tones with different discrete values of g.



(d) Loop filter magnitude responses for different discrete values of a when $g = 1$.

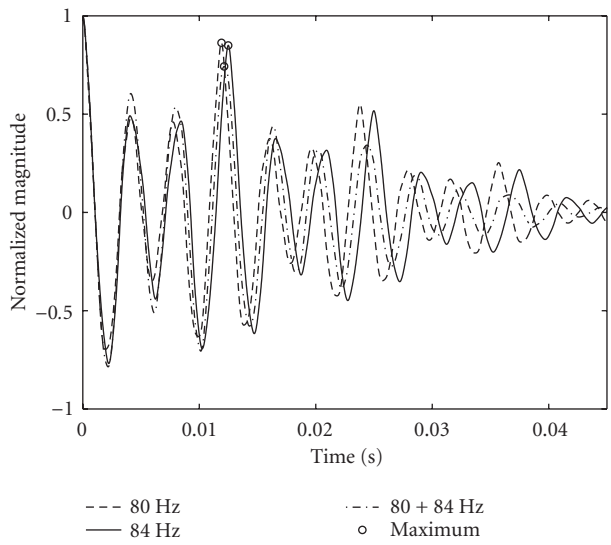
Figure 5: Discretizing the parameters g and a.

fundamental frequencies of two polarizations differ, the frequency estimate settles in the middle of the frequencies, as shown in Figure 6. Frequency discrimination thresholds as a function of frequency have been proposed in [33]. Also the audibility of beating and amplitude modulation has been studied in [27]. These results do not give us directly the discrimination thresholds for the difference in the fundamental frequencies of the two-polarization string model, because the fluctuation strength in an output sound depends on the fundamental frequencies and the decay parameters g and a.

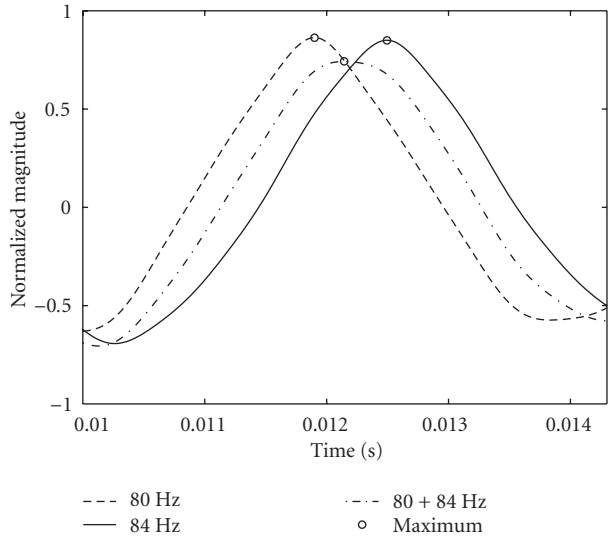
The sensitivity of parameters can be examined when a synthesized tone with known parameter values is used as a target tone with which another synthesized tone is compared. Varying one parameter after another and freezing the others, we obtain the error as a function of the parameters. In Figure 7, the target values of $f_{0,v}$ and $f_{0,h}$ are 331 and 330 Hz. The solid line shows the error when $f_{0,v}$ is linearly swept from 327 to 344 Hz. The global minimum is obviously found when $f_{0,v} = 331$ Hz. Interestingly, another nonzero local minimum is found when $f_{0,v} = 329$ Hz, that is, when the beating is similar. The dashed line shows the error when both $f_{0,v}$ and $f_{0,h}$ are varied but the difference in the fundamental frequencies is kept constant. It can be seen that the difference is more dominant than the absolute frequency value and have to be therefore discretized with higher resolution. Instead of operating the fundamental frequency parameters directly, we optimize the difference $d_f = |f_{0,v} - f_{0,h}|$ and the mean frequency $f_0' = |f_{0,v} + f_{0,h}|/2$ individually. Combining previous results from [27, 33] with our informal listening test, we have discretized d_f with 100 discrete values and f_0' with 20. The range of variation is set as follows:

$$r_p = \pm\left(\frac{\hat{f}_0}{10}\right)^{1/3}, \tag{16}$$

which is shown in Figure 8.



(a) Entire autocorrelation function.



(b) Zoomed around the maximum.

Figure 6: Three autocorrelation functions. Dashed and solid lines show functions for two single-polarization guitar tones with fundamental frequencies of 80 and 84 Hz. Dash-dotted line corresponds to a dual-polarization guitar tone with fundamental frequencies of 80 and 84 Hz.

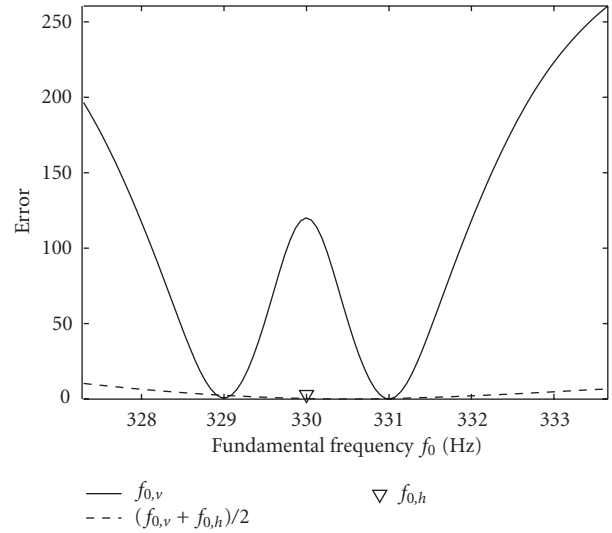


Figure 7: Error as a function of the fundamental frequencies. The target values of $f_{0,v}$ and $f_{0,h}$ are 331 and 330 Hz. The solid line shows the error when $f_{0,h} = 330$ and $f_{0,v}$ is linearly swept from 327 to 334 Hz. The dashed line shows the error when both frequencies are varied simultaneously while the difference remains similar.

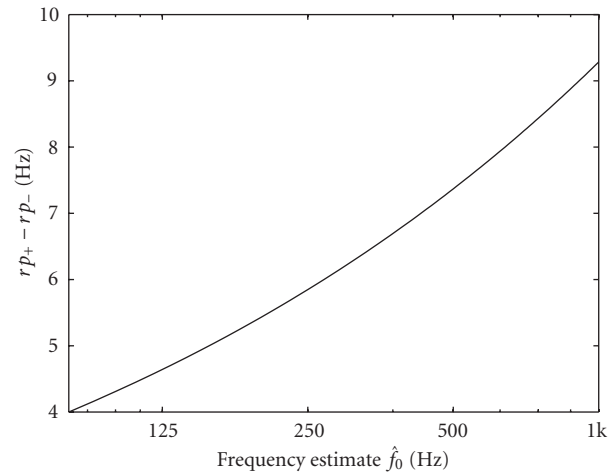


Figure 8: The range of variation in fundamental frequency as a function of frequency estimate from 80 to 1000 Hz.

5.3. Other parameters

The tolerances for the mixing coefficients m_p, m_o, and g_c have not been studied and the parameters have been earlier adjusted by trial and error [5]. Therefore, no initial guesses are made for these parameters. The sensitivities of the mixing coefficients are examined in an example case in Figure 9, where $m_p = 0.5$, $m_p = 0.5$, and $m_p = 0.1$. It can be seen that the parameters m_p and m_o are most sensitive near the boundaries and the parameter g_c is most sensitive near zero. Ranges for m_p and m_o are discretized with 40 values according to Figure 10. This method is applied to the parameter g_c, the range of which is limited to 0–0.5.

Discretizing the nine parameters this way results in 2.77×10^{15} combinations in total for a single tone. For an acoustic guitar, about 120 tones with different dynamic levels and playing styles have to be analyzed. It is obvious that an exhaustive search is out of question.

6. GENETIC ALGORITHM

GAs mimic the evolution of nature and take advantage of the principle of survival of the fittest [34]. These algorithms operate on a population of potential solutions improving

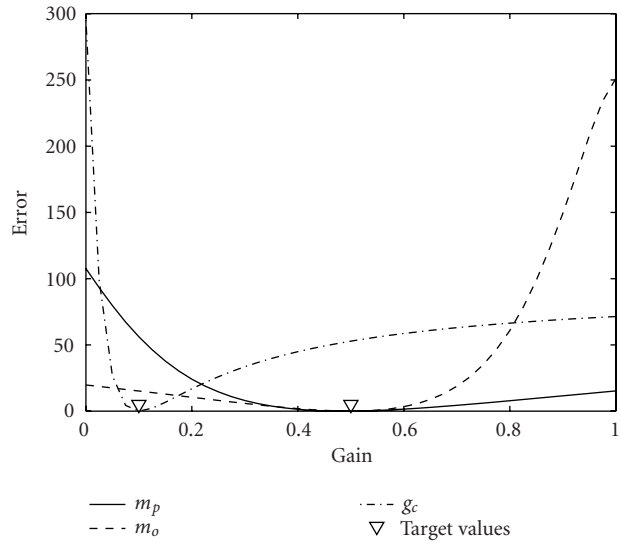


Figure 9: Error as a function of mixing coefficients m_p, m_o, and coupling coefficient g_c. Target values are $m_p = m_o = 0.5$ and $g_c = 0.1$.

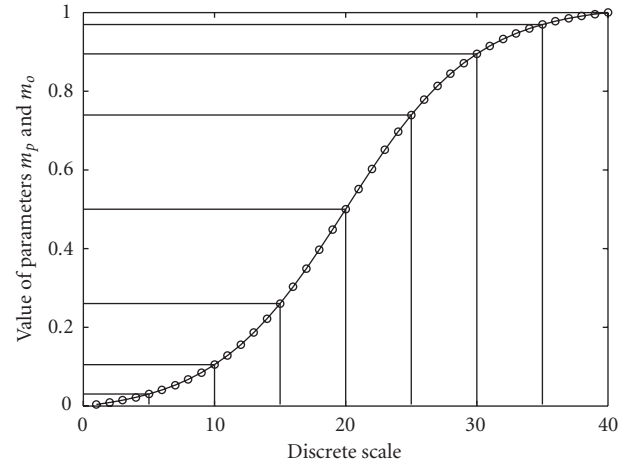


Figure 10: Discrete values for the parameters m_p and m_o.

characteristics of the individuals from generation to generation. Each individual, called a chromosome, is made up of an array of genes that contain, in our case, the actual parameters to be estimated.

In the original algorithm design, the chromosomes were represented with binary numbers [35]. Michalewicz [36] showed that representing the chromosomes with floating-point numbers results in faster, more consistent, higher precision, and more intuitive solution of the algorithm. We use a GA with the floating-point representation, although the parameter space is discrete, as discussed in Section 5. We have also experimented with the binary-number representation, but the execution time of the iteration becomes slow. Nonuniformly graduated parameter space is transformed into the uniform scales where the GA operates on. The floating-point numbers are rounded to the nearest discrete parameter value. The original floating-point operators are discussed in [36], where the characteristics of the operators are also described. Few modifications to the original mutation operators in step 5 have been made to improve the operation of the algorithm with the discrete grid.

The algorithm we use is implemented as follows.

(1) Analyze the recorded tone to be resynthesized using the analysis methods discussed in Section 3. The range of the parameter f_0' is chosen and the excitation signal is produced according to these results. Calculate the threshold of masking (Section 4) and the discrete scales for the parameters (Section 5).

(2) Initialization: create a population of S_p individuals (chromosomes). Each chromosome is represented as a vector array $\vec{x}$, with nine components (genes), which contains the actual parameters. The initial parameter values are randomly assigned.

(3) Fitness calculation: calculate the perceptual fitness of each individual in the current population according to (15).

(4) Selection of individuals: select individuals from the current population to produce the next generation based upon the individual's fitness. We use the normalized geometric selection scheme [37], where the individuals are first ranked according to their fitness values. The probability of selecting the ith individual to the next generation is then calculated by

$$P_i = q'(1-q)^{r-1}, \tag{17}$$

where

$$q' = \frac{q}{1-(1-q)^{S_p}}, \tag{18}$$

q is the user-defined parameter which denotes the probability of selecting the best individual, and r is the rank of the individual, where 1 is the best and S_p is the worst. Decreasing the value of q slows the convergence.

(5) Crossover: randomly pick a specified number of parents from selected individuals. An offspring is produced by crossing the parents with a simple, arithmetical, and heuristic crossover scheme. Simple crossover creates two new individuals by splitting the parents in a random point and swapping the parts. Arithmetical crossover produces two linear combinations of the parents with a random weighting. Heuristic crossover produces a single offspring $\vec{x}_o$ which is a linear extrapolation of the two parents $\vec{x}_{p,1}$ and $\vec{x}_{p,2}$ as follows:

$$\vec{x}_o = h(\vec{x}_{p,2} - \vec{x}_{p,1}) + \vec{x}_{p,2}, \tag{19}$$

where $0 \le h \le 1$ is a random number and the parent $\vec{x}_{p,2}$ is not worse than $\vec{x}_{p,1}$. Nonfeasible solutions are possible and if no solution is found after w attempts, the operator gives no offspring. Heuristic crossover contributes to the precision of the final solution.

(6) Mutation: randomly pick a specified number of individuals for mutation. Uniform, nonuniform, multi-nonuniform, and boundary mutation schemes are used. Mutation works with a single individual at a time. Uniform mutation sets a randomly selected parameter (gene) to a uniform random number between the boundaries. Nonuniform mutation operates uniformly at early stage and more locally as the current generation approaches the maximum generation. We have defined the scheme to operate in such a way that the change is always at least one discrete step. The degree of nonuniformity is controlled with the parameter b. Nonuniformity is important for fine-tuning. Multi-nonuniform mutation changes all of the parameters in the current individual. Boundary mutation sets a parameter to one of its boundaries and is useful if the optimal solution is supposed to lie near the boundaries of the parameter space. The boundary mutation is used in special cases, such as staccato tones.

(7) Replace the current population with the new one.

(8) Repeat steps 3, 4, 5, 6, and 7 until termination.

Our algorithm is terminated when a specified number of generations is produced. The number of generations defines the maximum duration of the algorithm. In our case, the time spent with the GA operations is negligible compared to the synthesis and fitness calculation. Synthesis of a tone with candidate parameter values takes approximately 0.5 second, while the duration of the error calculation is 1.2 second. This makes 1.7 second in total for a single parameter set.

7. EXPERIMENTATION AND RESULTS

To study the efficiency of the proposed method, we first tried to estimate the parameters for the sound produced by the synthesis model itself. First, the same excitation signal extracted from a recorded tone by the method described in [24] was used for target and output sounds. A more realistic case is simulated when the excitation for resynthesis is extracted from the target sound. The system was implemented with Matlab software and all runs were performed on an Intel Pentium III computer. We used the following parameters for all experiments: population size $S_p = 60$, number of generations $= 400$, probability of selecting the best individual $q = 0.08$, degree of nonuniformity $b = 3$, retries $w = 3$, number of crossovers $= 18$, and number of mutations $= 18$.

The *pitch synchronous Fourier transform* scheme, where the window length L_w is synchronized with the period length of the signal such that $L_w = 4f_s/f_0$, is utilized in this work. The overlap of the used hanning windows is 50%, implying that hop size $H = L_w/2$. The sampling rate is $f_s = 44100\,\text{Hz}$ and the length of FFT is $N = 2048$.

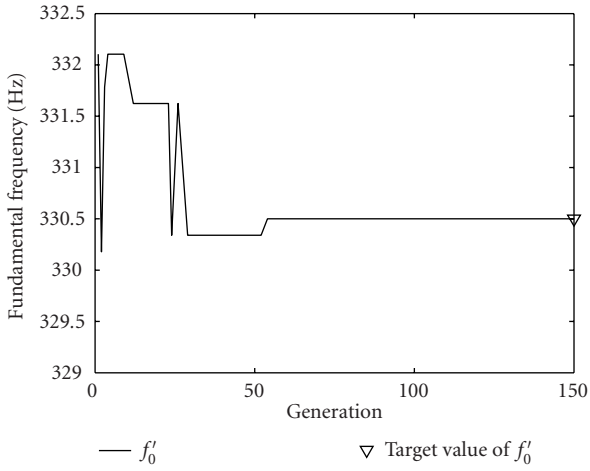
The original and the estimated parameters for three experiments are shown in Table 2. In experiment 1 the original excitation is used for the resynthesis. The exact parameters are estimated for the difference d_f and for the decay parameters g_h, g_v, and a_v. The adjacent point in the discrete grid is estimated for the decay parameter a_h. As can be seen in Figure 7, the sensitivity of the mean frequency is negligible compared to the difference d_f, which might be the cause of deviations in mean frequency. Differences in the mixing parameters m_o, m_p, and the coupling coefficient g_c can be noticed. When running the algorithm multiple times, no explicit optima for mixing and coupling parameters were found. However, synthesized tones produced by corresponding parameter values are indistinguishable. That is to say that the parameters m_p, m_o, and g_c are not orthogonal, which is clearly a problem with the model and also impairs the efficiency of our parameter estimation algorithm.

To overcome the nonorthogonality problem, we have run the algorithm with constant values of $m_p = m_o = 0.5$ in experiment 2. If the target parameters are set according to discrete grid, the exact parameters with zero error are estimated. The convergence of the parameters and the error of such case is shown in Figure 11. Apart from the fact that the parameter values are estimated precisely, the convergence of the algorithm is very fast. Zero error is already found in generation 87.

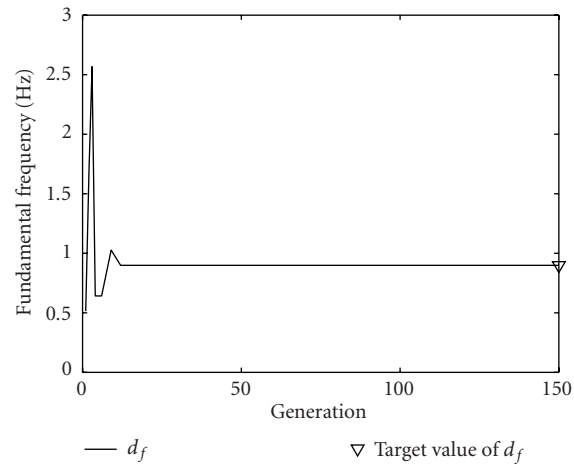
A similar behavior is noticed in experiment 3 where an extracted excitation is used for resynthesis. The difference and the decay parameters g_h and g_v are again estimated precisely. Parameters m_p, m_o, and g_c drift as in previous experiment. Interestingly, $m_p = 1$, which means that the straight path to vertical polarization is totally closed. The model is, in a manner of speaking, rearranged in such a way that the individual string models are in series as opposed to the original construction where the polarization are arranged in parallel.

Unlike in experiments 1 and 2, the exact parameter values are not so relevant since different excitation signals are used for the target and estimated tones. Rather than looking into the parameter values, it is better to analyze the tones produced with the parameters. In Figure 12, the overall temporal envelopes and the envelopes of the first eight partials for the target and for the estimated tone are presented. As can be seen, the overall temporal envelopes are almost identical and the partial envelopes match well. Only the beating amplitude differs slightly but it is inaudible. This indicates that the parametrization of the model itself is not the best possible since similar tones can be synthesized with various parameter sets.

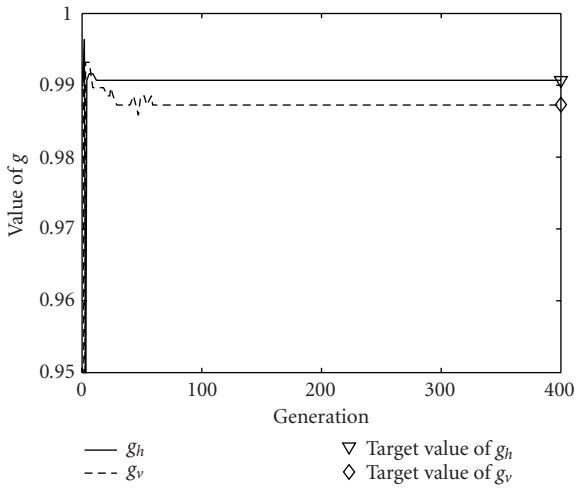
Our estimation method is designed to be used with real recorded tones. Time and frequency analysis for such case is shown in Figure 13. As can be seen, the overall temporal envelopes and the partial envelopes for a recorded tone are very similar to those that are analyzed from a tone that uses estimated parameter values. Appraisal of the perceptual quality of synthesized tones is left as a future project, but our informal listening indicates that the quality is comparable with or better than our previous methods and it does not require any hand tuning after the estimation procedure. Sound clips demonstrating these experiments are available at http://www.acoustics.hut.fi/publications/papers/jasp-ga.



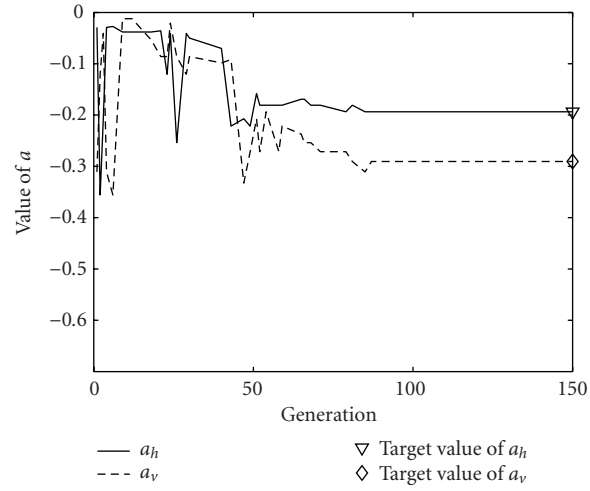
(a) Convergence of the parameter f_0'.



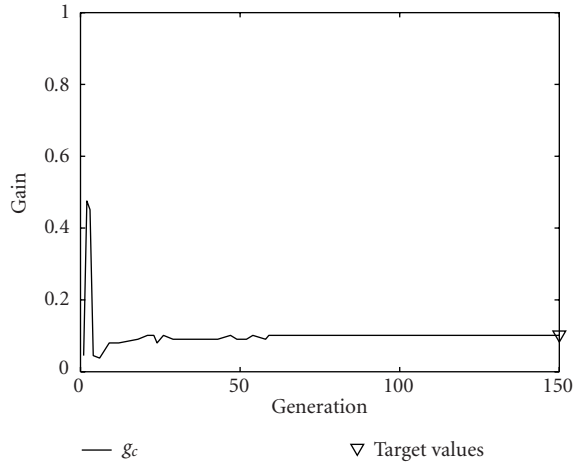
(b) Convergence of the parameter d_f.



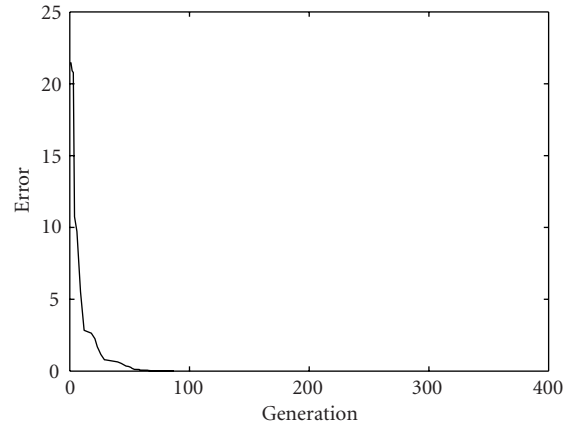
(c) Convergence of the parameters g_h and g_v.



(d) Convergence of the parameters a_h and a_v.



(e) Convergence of the parameter g_c.

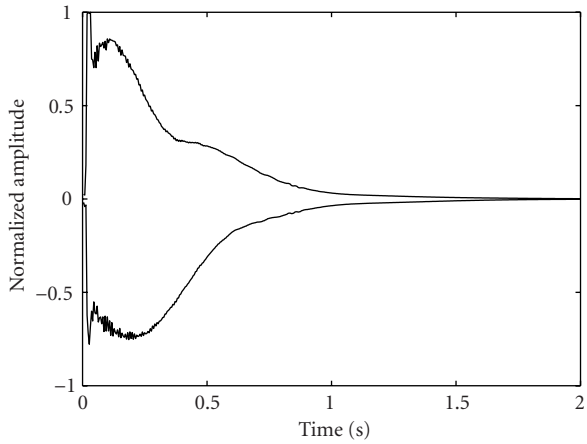


(f) Convergence of the error.

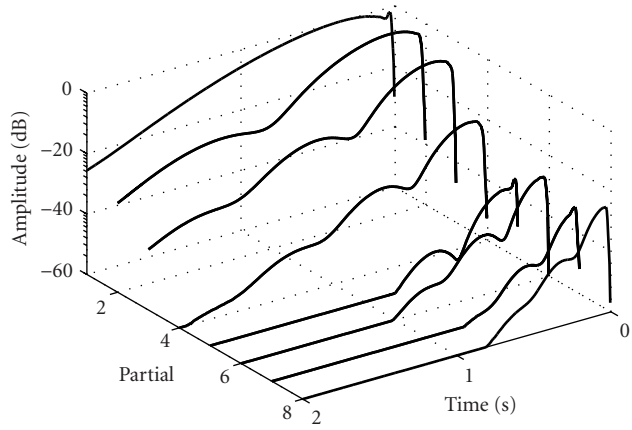
FIGURE 11: Convergence of the seven parameters and the error for experiment 2 in Table 2. Mixing coefficients are frozen as $m_p = m_o = 0.5$ to overcome the nonorthogonality problem. One hundred and fifty generations are shown and the original excitation is used for the resynthesis.

Table 2: Original and estimated parameters when a synthesized tone with known parameter values are used as a target tone. The original excitation is used for resynthesis in experiments 1 and 2 and the extracted excitation is used for the resynthesis in experiment 3. In experiment 2 the mixing coefficients are frozen as $m_p = m_o = 0.5$.

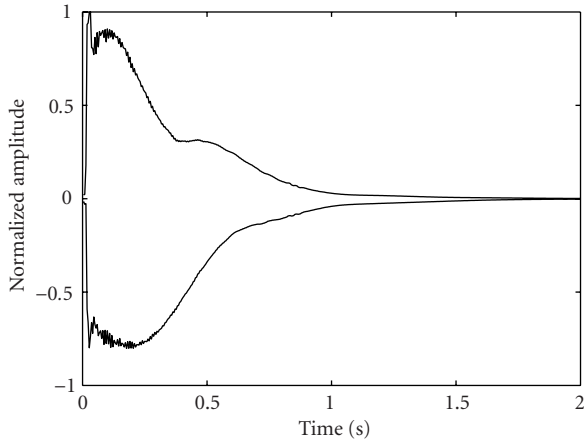
Parameter	Target parameter	Experiment 1	Experiment 2	Experiment 3
f_0'	330.5409	331.000850	330.5409	330.00085
d_f	0.8987	0.8987	0.8987	0.8987
g_h	0.9873	0.9873	0.9873	0.9873
a_h	−0.2905	−0.3108	−0.2905	−0.2071
g_v	0.9907	0.9907	0.9907	0.9907
a_v	−0.1936	−0.1936	−0.1936	−0.1290
m_p	0.5	0.2603	(0.5)	1.000
m_o	0.5	0.6971	(0.5)	0.8715
g_c	0.1013	0.2628	0.1013	0.2450
Error	—	0.0464	0	0.4131



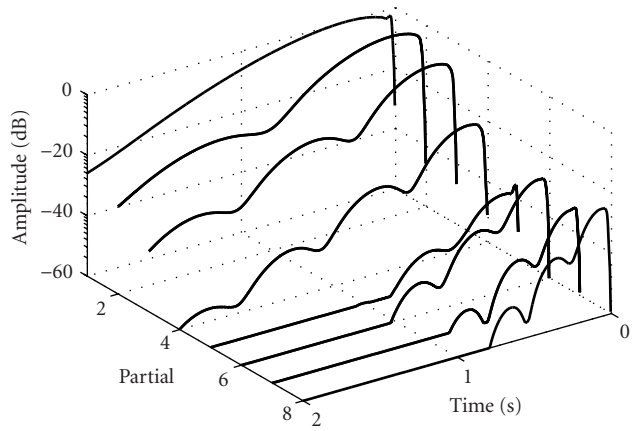
(a) Overall temporal envelope for a target tone.



(b) First eight partials for a target tone.

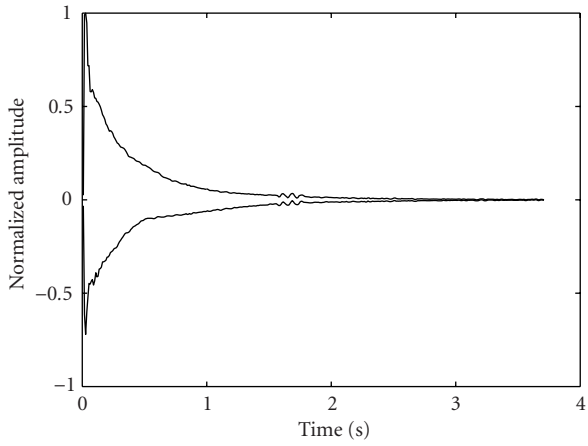


(c) Overall temporal envelope for an estimated tone.

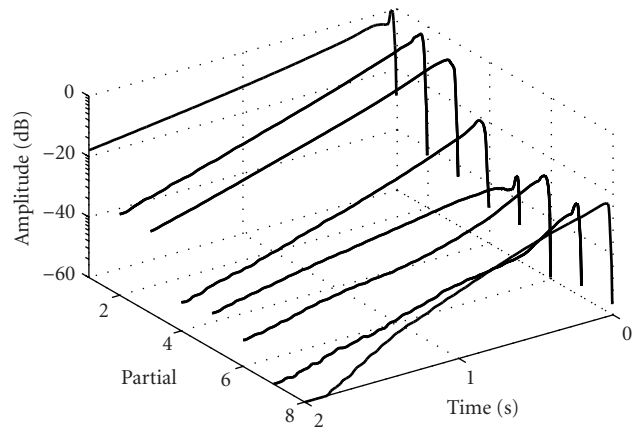


(d) First eight partials for an estimated tone.

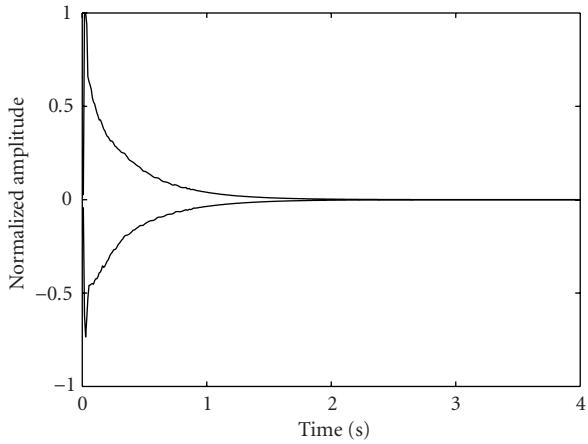
Figure 12: Time and frequency analysis for experiment 3 in Table 2. The synthesized target tone is produced with known parameter values and the synthesized tone uses estimated parameter values. Extracted excitation is used for the resynthesis.



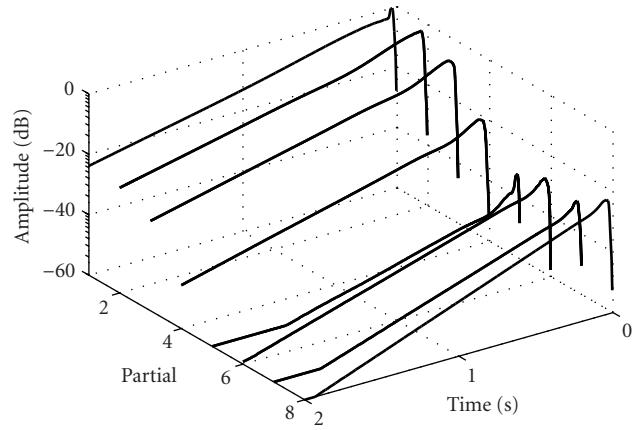
(a) Waveform for a recorded tone.



(b) First eight partials for a recorded tone.



(c) Waveform for an estimated tone.



(d) First eight partials for an estimated tone.

Figure 13: Time and frequency analysis for a recorded tone and for a synthesized tone that uses estimated parameter values. Extracted excitation is used for the resynthesis. Estimated parameter values are $f_0' = 331.1044$, $d_f = 1.1558$, $g_h = 0.9762$, $a_h = -0.4991$, $g_v = 0.9925$, $a_v = 0.0751$, $m_p = 0.1865$, $m_o = 0.7397$, and $g_c = 0.1250$.

8. CONCLUSIONS AND FUTURE WORK

A parameter estimation scheme based on a GA with a perceptual fitness function was designed and tested for a plucked string synthesis algorithm. The synthesis algorithm is used for natural-sounding synthesis of various string instruments. For this purpose, automatic parameter estimation is needed. Previously, the parameter values have been extracted from recordings using more traditional signal processing techniques, such as short-term Fourier transform, linear regression, and linear digital filter design. Some of the parameters could not have been reliably estimated from the recorded sound signal, but they have had to be fine-tuned manually by an expert user.

In this work, we presented a fully automatic parameter extraction method for string synthesis. The fitness function we use employs knowledge of properties of the human auditory system, such as frequency-dependent sensitivity and frequency masking. In addition, a discrete parameter space has been designed for the synthesizer parameters. The range, the nonuniformity of the sampling grid, and the number of allowed values for each parameter were chosen based on former research results, experiments on parameter sensitivity, and informal listening.

The system was tested with both synthetic and real tones. The signals produced with the synthesis model itself are considered a particularly useful class of test signals because there will always be a parameter set that exactly reproduces the analyzed signal (although discretization of the parameter space may limit the accuracy in practice). Synthetic signals offered an excellent tool to evaluate the parameter estimation procedure, which was found to be accurate with two choices of excitation signal to the synthesis model. The quality of resynthesis of real recordings is more difficult to measure as there are no known correct parameter values. As high-quality synthesis of several plucked string instrument sounds has been possible in the past with the same synthesis algorithm, we

expected to hear good results using the GA-based method, which was also the case.

Appraisal of synthetic tones that use parameter values from the proposed GA-based method is left as a future project. Listening tests similar to those used for evaluating high-quality audio coding algorithms may be useful for this task.

REFERENCES

[1] J. O. Smith, "Physical modeling using digital waveguides," *Computer Music Journal*, vol. 16, no. 4, pp. 74–91, 1992.

[2] J. O. Smith, "Efficient synthesis of stringed musical instruments," in *Proc. International Computer Music Conference (ICMC '93)*, pp. 64–71, Tokyo, Japan, September 1993.

[3] M. Karjalainen, V. Välimäki, and Z. Jánosy, "Towards high-quality sound synthesis of the guitar and string instruments," in *Proc. International Computer Music Conference (ICMC '93)*, pp. 56–63, Tokyo, Japan, September 1993.

[4] V. Välimäki, J. Huopaniemi, M. Karjalainen, and Z. Jánosy, "Physical modeling of plucked string instruments with application to real-time sound synthesis," *Journal of the Audio Engineering Society*, vol. 44, no. 5, pp. 331–353, 1996.

[5] M. Laurson, C. Erkut, V. Välimäki, and M. Kuuskankare, "Methods for modeling realistic playing in acoustic guitar synthesis," *Computer Music Journal*, vol. 25, no. 3, pp. 38–49, 2001.

[6] G. Weinreich, "Coupled piano strings," *Journal of the Acoustical Society of America*, vol. 62, no. 6, pp. 1474–1484, 1977.

[7] M. Karjalainen, V. Välimäki, and T. Tolonen, "Plucked-string models: from the Karplus-Strong algorithm to digital waveguides and beyond," *Computer Music Journal*, vol. 22, no. 3, pp. 17–32, 1998.

[8] T. Tolonen and V. Välimäki, "Automated parameter extraction for plucked string synthesis," in *Proc. International Symposium on Musical Acoustics (ISMA '97)*, pp. 245–250, Edinburgh, Scotland, August 1997.

[9] C. Erkut, V. Välimäki, M. Karjalainen, and M. Laurson, "Extraction of physical and expressive parameters for model-based sound synthesis of the classical guitar," in *the Audio Engineering Society 108th International Convention*, Paris, France, February 2000, preprint 5114, http://lib.hut.fi/Diss/2002/isbn9512261901.

[10] A. Nackaerts, B. De Moor, and R. Lauwereins, "Parameter estimation for dual-polarization plucked string models," in *Proc. International Computer Music Conference (ICMC '01)*, pp. 203–206, Havana, Cuba, September 2001.

[11] S.-F. Liang and A. W. Y. Su, "Recurrent neural-network-based physical model for the chin and other plucked-string instruments," *Journal of the Audio Engineering Society*, vol. 48, no. 11, pp. 1045–1059, 2000.

[12] C. Drioli and D. Rocchesso, "Learning pseudo-physical models for sound synthesis and transformation," in *Proc. IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1085–1090, San Diego, Calif, USA, October 1998.

[13] V.-V. Mattila and N. Zacharov, "Generalized listener selection (GLS) procedure," in *the Audio Engineering Society 110th International Convention*, Amsterdam, The Netherlands, 2001, preprint 5405.

[14] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, 1990.

[15] A. Härmä, M. Karjalainen, L. Savioja, V. Välimäki, U. Laine, and J. Huopaniemi, "Frequency-warped signal processing for audio applications," *Journal of the Audio Engineering Society*, vol. 48, no. 11, pp. 1011–1031, 2000.

[16] J. Vuori and V. Välimäki, "Parameter estimation of non-linear physical models by simulated evolution—application to the flute model," in *Proc. International Computer Music Conference (ICMC '93)*, pp. 402–404, Tokyo, Japan, September 1993.

[17] A. Horner, J. Beauchamp, and L. Haken, "Machine tongues XVI: Genetic algorithms and their application to FM matching synthesis," *Computer Music Journal*, vol. 17, no. 4, pp. 17–29, 1993.

[18] R. Garcia, "Automatic generation of sound synthesis techniques," M.S. thesis, Massachusetts Institute of Technology, Cambridge, Mass, USA, 2001.

[19] C. Johnson, "Exploring the sound-space of synthesis algorithms using interactive genetic algorithms," in *Proc. AISB Workshop on Artificial Intelligence and Musical Creativity*, pp. 20–27, Edinburgh, Scotland, April 1999.

[20] D. Jaffe and J. O. Smith, "Extensions of the Karplus-Strong plucked-string algorithm," *Computer Music Journal*, vol. 7, no. 2, pp. 56–69, 1983.

[21] C. Erkut, M. Laurson, M. Kuuskankare, and V. Välimäki, "Model-based synthesis of the ud and the renaissance lute," in *Proc. International Computer Music Conference (ICMC '01)*, pp. 119–122, Havana, Cuba, September 2001.

[22] C. Erkut and V. Välimäki, "Model-based sound synthesis of tanbur, a Turkish long-necked lute," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, pp. 769–772, Istanbul, Turkey, June 2000.

[23] C. Roads, *The Computer Music Tutorial*, MIT Press, Cambridge, Mass, USA, 1996.

[24] V. Välimäki and T. Tolonen, "Development and calibration of a guitar synthesizer," *Journal of the Audio Engineering Society*, vol. 46, no. 9, pp. 766–778, 1998.

[25] C. Erkut, M. Karjalainen, P. Huang, and V. Välimäki, "Acoustical analysis and model-based sound synthesis of the kantele," *Journal of the Acoustical Society of America*, vol. 112, no. 4, pp. 1681–1691, 2002.

[26] B. Bank, "Physics-based sound synthesis of the piano," Tech. Rep. 54, Helsinki University of Technology, Laboratory of Acoustics and Audio Signal Processing, Espoo, Finland, May 2000, http://www.acoustics.hut.fi/publications/2000.html.

[27] E. Zwicker and H. Fastl, *Psychoacoustics: Facts and Models*, Springer-Verlag, Berlin, Germany, 1990.

[28] M. Lagrange and S. Marchand, "Real-time additive synthesis of sound by taking advantage of psychoacoustics," in *Proc. COST-G6 Conference on Digital Audio Effects (DAFx '01)*, pp. 5–9, Limerick, Ireland, December 2001.

[29] C. W. Wun and A. Horner, "Perceptual wavetable matching for synthesis of musical instrument tones," *Journal of the Audio Engineering Society*, vol. 49, no. 4, pp. 250–262, 2001.

[30] J. D. Johnston, "Transform coding of audio signals using perceptual noise criteria," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 2, pp. 314–323, 1988.

[31] M. R. Schroeder, B. S. Atal, and J. L. Hall, "Optimizing digital speech coders by exploiting masking properties of the human ear," *Journal of the Acoustical Society of America*, vol. 66, no. 6, pp. 1647–1652, 1979.

[32] H. Järveläinen and T. Tolonen, "Perceptual tolerances for decay parameters in plucked string synthesis," *Journal of the Audio Engineering Society*, vol. 49, no. 11, pp. 1049–1059, 2001.

[33] C. C. Wier, W. Jesteadt, and D. M. Green, "Frequency discrimination as a function of frequency and sensation level," *Journal of the Acoustical Society of America*, vol. 61, no. 1, pp. 178–184, 1977.

[34] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, Mass, USA, 1998.

[35] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Mich, USA, 1975.
[36] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, AI Series. Springer-Verlag, New York, NY, USA, 1992.
[37] J. Joines and C. Houck, "On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's," in *IEEE International Symposium on Evolutionary Computation*, pp. 579–584, Orlando, Fla, USA, June 1994.

Janne Riionheimo was born in Toronto, Canada, in 1974. He studies acoustics and digital signal processing at Helsinki University of Technology, Espoo, Finland, and music technology, as a secondary subject, at the Centre for Music and Technology, Sibelius Academy, Helsinki, Finland. He is currently finishing his M.S. thesis, which deals with parameter estimation of a physical synthesis model. He has worked as a Research Assistant at the HUT Laboratory of Acoustics and Audio Signal Processing from 2001 until 2002. His research interests include physical modeling of musical instruments and musical acoustics. He is also working as a Recording Engineer.

Vesa Välimäki was born in Kuorevesi, Finland, in 1968. He received his Master of Science in Technology, Licentiate of Science in Technology, and Doctor of Science in Technology degrees, all in electrical engineering from Helsinki University of Technology (HUT), Espoo, Finland, in 1992, 1994, and 1995, respectively. Dr. Välimäki worked at the HUT Laboratory of Acoustics and Audio Signal Processing from 1990 until 2001. In 1996, he was a Postdoctoral Research Fellow in the University of Westminster, London, UK. He was appointed Docent in audio signal processing at HUT in 1999. During the academic year 2001–2002 he was Professor of Signal Processing at Pori School of Technology and Economics, Tampere University of Technology, Pori, Finland. In August 2002, he returned to HUT, where he is currently Professor of Audio Signal Processing. His research interests are in the application of digital signal processing to audio and music. He has published more than 120 papers in international journals and conferences. He holds two patents. Dr. Välimäki is a senior member of the IEEE Signal Processing Society and a member of the Audio Engineering Society and the International Computer Music Association.