GA-BASED APPROACH TO PITCH RECOGNITION OF MUSICAL CONSONANCE

Masanori Natsui, Shunichi Kubo and Yoshiaki Tadokoro
Dept. of Information and Computer Sciences, Toyohashi University of Technology
1-1 Hibarigaoka, Tempaku-cho, Toyohashi-shi, Aichi 441-8580, Japan

Keywords: Pitch recognition, Genetic algorithm, Musical consonance, Template matching.

Abstract: This paper presents a novel method for the pitch recognition of the musical consonance (i.e., unison or octave) using genetic algorithm (GA). GA is a kind of optimization techniques based on natural selection and genetics. In our method, the pitch recognition is performed by the following two-step procedure: (i) search space reduction using the comb filter estimation, and (ii) evolutionary parameter estimation of tone parameters such as notes and volumes by minimizing error between a target waveform and a synthesized waveform using sound templates with estimated parameters. The potential capability of the system is demonstrated through the pitch estimation of randomly-generated consonances. Experimental results show that the system can successfully estimate chords with more than 84% success rate for two-note consonances, and more than 71% success rate for three-note consonances.

1 INTRODUCTION

Automatic music transcription is important for many applications including music archival, music retrieval, supports of music composition/arrangement, and also significant problems in machine perception(Sterian and Wakefield, 2000; Pollastri, 2002; Roads, 1985; Roads, 1996; Piszczalski and Galler, 1977). The study of automatic musical transcription can be classified into some categories, and that of the pitch detection is the most important task and many studies have been done. Most of old studies are for monophony, and based on the spectrum analysis using the fast Fourier transform (FFT). On the other hand, the novel technologies such as neural network, fuzzy logic, and hidden Marcov model have also been proposed in the recent studies(Klapuri, 2003).

For the pitch estimation of polyphonic sounds, we have proposed a unique method based on comb filters ($H(z) = 1 - z^{-N}$)(Tadokoro and Yamaguchi, 2001; Tadokoro et al., 2002; Tadokoro et al., 2003). The comb filter can eliminate a fundamental frequency and its harmonic components of a sound by simple subtraction. So far, we have presented that cascade or parallel connections of the comb filters enable the polyphonic pitch estimation and can be effective for the realization of the automatic music transcription system.

A difficult problem in the polyphonic pitch estimation is that some frequency components of one note may be overlapped with harmonics of other notes. In fact, composers often use chords containing notes that have a simple ratio between their fundamental frequencies, such as 1:1 (perfect unison), 2:1 (perfect octave), or 3:2 (perfect fifth), since these codes called *consonances* typically produce sounds which are pleasing to the human ear. If one note having a fundamental frequency of fHz and another note having that of $2f$Hz are produced at the same time, then every harmonic of the upper note will be overlapped to the even harmonics of the lower note (Fig. 1). To infer the presence of the upper note, we have to use some other information which is obtained by a technique except traditional methods such as spectrum analysis.

From this viewpoint, we propose a unique method of the pitch estimation based on *genetic algorithm* (GA). GA is an optimization algorithm based on a model of evolution in life. In this paper, we demonstrate the possibility of the GA-based pitch estimation method through the experimental pitch estimation of musical consonance. The key ideas presented here are: (i) time-domain template matching based on GA, and (ii) search space reduction using the pitch estima-

Natsui M., Kubo S. and Tadokoro Y. (2006).
GA-BASED APPROACH TO PITCH RECOGNITION OF MUSICAL CONSONANCE.
In *Proceedings of the Third International Conference on Informatics in Control, Automation and Robotics*, pages 47-52
DOI: 10.5220/0001209400470052

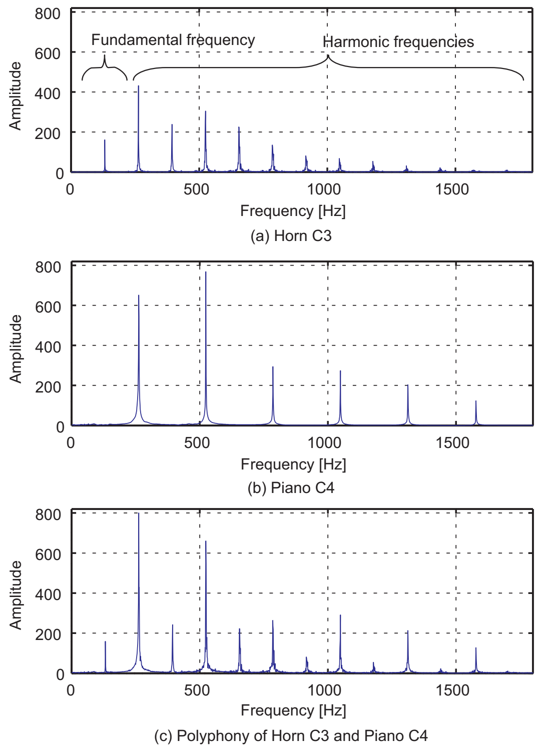

Figure 1: Spectra of music instruments: (a) Horn C3, (b) Piano C4, and (c) polyphony of the two tones.

```
program Genetic Algorithm;
 begin
  t := 0;
  {t: Number of generations.}
  initialize(P(t));
  {P(t): Population.}
  evaluate(P(t));
  while t ≤ Max. num. of gen. do
   begin
    C(t) := crossover(P(t));
    M(t) := mutation(P(t));
    evaluate(C(t) ∪ M(t));
    P(t + 1) := select(C(t) ∪ M(t) ∪ P(t));
    t := t + 1;
   end
 end.
```

Figure 2: Typical flow of GA.

tion result based on the comb filetrs.

This paper is organized as follows: Section 2 presents the basic concept of the pitch estimation system using GA. Section 3 shows an overview of the proposed pitch estimation system. Section 4 demonstrates the experimental result of the pitch estimation. Section 5 is the conclusion and future prospects.

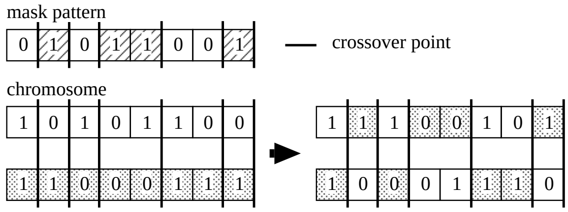

Figure 3: Uniform crossover.

Table 1: Instruments and pitches stored in the sound template.

Instrument	Pitch
Alt Saxophone (AS)	D3A5
Clarinet (CL)	D3B5
Horn (HR)	C3F5
Trumpet (TR)	E3B5
Viora (VL)	C3B5
Violin (VN)	G3B5
Piano (PF)	C3B5

2 GENETIC ALGORITHM

Genetic algorithm (GA) can be regarded as a unique variation of evolutionary computation techniques (Back et al., 1997; Holland, 1975; Goldberg, 1989). In general, evolutionary methods mimic the process of natural evolution, the driving process for emergence of complex structures well-adapted to the given environment. The better an individual performs under the conditions the greater is the chance for the individual to live for a longer while and generate offspring. As a result, the individuals are transformed to the suitable forms on the designer's defined constraint.

Figure 2 shows the overall procedure of GA. At first, GA generates embryonic individuals randomly to form the initial population $P(0)$. In the traditional GA, each individual is represented by a fixed-length bit string. The next step is to evaluate a fitness function at all individuals in $P(t)$. A value for fitness is assigned to each individual depending on how close it actually is to solving the problem. After the evaluation, the system selects a set of individuals having higher fitness values to perform evolutionary operations: *crossover* and *mutation*. The crossover recombines two individuals into two new bit strings. The mutation operation, on the other hand, flips the values of chosen bits to their complements. There are many ways how to do crossover and mutation. For example, Figure 3 shows an example of crossover operation called *uniform crossover*.

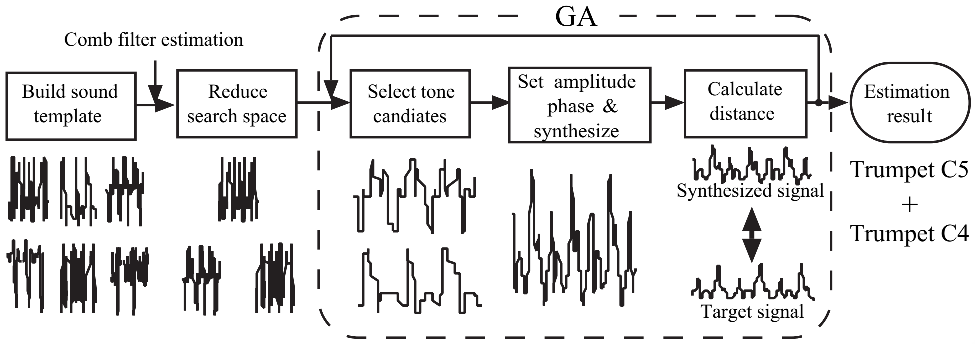

Figure 4: Overview of the proposed system.

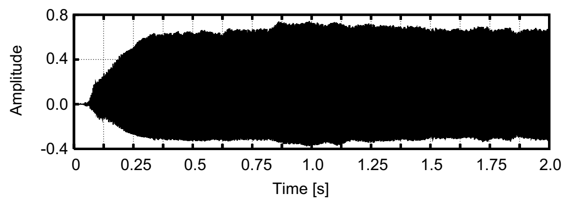

Figure 5: Sound waveform (Trumpet, C4).

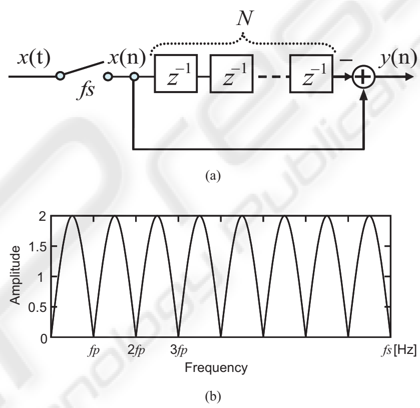

Figure 6: Notch-type comb filter: (a) block diagram, and (b) frequency characteristic.

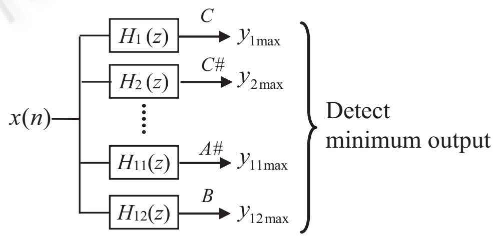

Figure 7: Parallel-connected comb filters for pitch estimation.

3 PITCH ESTIMATION SYSTEM USING GA

Figure 4 shows the overview of the proposed pitch estimation system. At first, we make a database which contains waveforms of sound templates. Then the system determines waveforms for the estimation by using the result of comb-filter-based pitch estimation. Finally, the system applies GA to search the optimal parameters such as amplitudes and phases, which minimize the squared error between a target waveform and a generated one. A more detailed description is provided in the following.

3.1 Sound Template

We make a database which contains a waveform of each instrument and tone shown in Table 1. In this experiment, the RWC music database(Goto, 2004) is used as the original data. We use a waveform appeared after 500ms from the beginning of a sound data, which is considered to have the level of the steady state amplitude (*sustain*), as a sound template. The sampling rate is 44.1 kHz, and the length of each template is 54ms (2385 points). Note that the maximum amplitude of each template is normalized to 1.0.

3.2 Search Space Reduction Using Comb Filter Estimation

The transfer function of the notch-type comb filter for a tone p is written by

$$H_p(z) = 1 - z^{-N_p}, N_p = [f_s/f_p],$$

where f_s is a sampling frequency, and f_p is a fundamental frequency for the tone p. The block diagram and its frequency characteristic are shown in Fig. 6 (a) and (b), respectively. The spectrum of a single note from a musical instrument usually has a set of

Table 2: Target parameters.

Paramater	Name	Range	Bit
Instrument and note	$note_k$	$1 \sim T^*$	$\lceil T/2 \rceil$
Amplitude	amp_k	$0 \sim 1.0$	8
Phase	$phase_k$	$0 \sim 2\pi$	6

(T: Number of templates after the search space reduction)

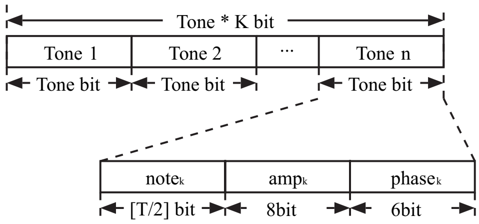

Figure 8: Individual representation.

peaks at harmonic ratios. That is, if the fundamental frequency is f_p, there are peaks at f_p, and also at $2f_p$, $3f_p$, $4f_p$, etc. Consequently, the operation of $H_p(z)$ eliminates all frequency components of the target tone at once.

As in the case of usual methods based on spectrum analysis, the comb filtering cannot separate the overlapped frequency components. For example, let a polyphony be composed of C3, E3, G3, and C4. We can estimate only the presence of C3, E3, and G3 by using a parallel-connected comb filters (Fig. 7), while the note C4 cannot be detected since every harmonic of the note C4 is overlapped to the even harmonics of the note C3. However, if we know the input sound is a chord of four notes, we can also estimate that a fundamental frequency of another tone is equal to the one of the known three tones. That is, the search space of sound templates can be reduced to the one which contains only the following tones:

Harmonics of C3: C4, G4, C5, E5,

Harmonics of E3: E4, B4, E5, G#5,

Harmonics of G3: G4, D4, G5, B5.

It achieves a reduction in the size of the search space and an improvement in the search efficiency.

3.3 Parameter Optimization Using GA

Figure 8 shows an individual representation in the system, which corresponds to a unique polyphonic sound. Here K is the number of tones included in a target polyphony, and amp_k, $note_k$, and $phase_k$ are parameters for kth notes, which have bit lengths and ranges of values shown in Table 2.

Table 3: Main parameter values for GA.

Parameter	Value
Maximum number of generations	250
Population size	750
Crossover method	Uniform
Crossover rate	0.8
Mutation rate	0.05
Mutation method	Uniform

At the beginning of a new evolutionary run, the system creates a set of randomly generated bit strings with above data format to form the initial population. A generated bit string is interpreted as its corresponding waveform defined as follows:

$$x'(n) = \sum_{k=1}^{K} \sum_{n=1}^{N} amp_k \cdot x_{note_k}(n + phase_k \frac{f_s}{f_k \cdot 2\pi}),$$

where $x_{note_k}(n)$ is a waveform of template $note_k$, N is a window size, f_s is a sampling frequency, and f_k is a fundamental frequency of a template $note_k$. Through an evolutionary run, the system searches the optimal waveform which minimizes the error as follows:

$$error = \sum_{n=1}^{N} \{x(n) - x'(n)\}^2 ,$$

where $x(n)$ is a target waveform.

4 EXPERIMENT

We demonstrate the potential capability of the GA-based pitch estimation of musical consonance. Note that the input sounds used in this experiment are polyphony which contain at least one consonance whose fundamental frequency ratio can be simply represented by $1 : n$ (n: integer). Table 3 summarizes the system parameters in this experiment. A set of evolutionary runs were carried out on a Linux PC (CPU: Intel Xeon 2.8GHz dual, RAM: 2GByte).

Figure 11 shows a fitness transition of the error between a generated waveform and a target one. Target sounds considered here are polyphony of randomly-selected three tones. We can see the staircase improvements of the best individual fitness on average.

Figures 9 and 10 depict an example snapshot of the population and a waveform of the best fitness individual on each generation. In Fig. 9, the horizontal axes indicate types of music instruments and pitches of tones, and the vertical axis indicates the number of individuals which contains a certain instrument and pitch. The input sound considered here is a polyphony of Trumpet C4, Alto Saxophone E4, and Horn C5. Given the initial random generation, the evolution is

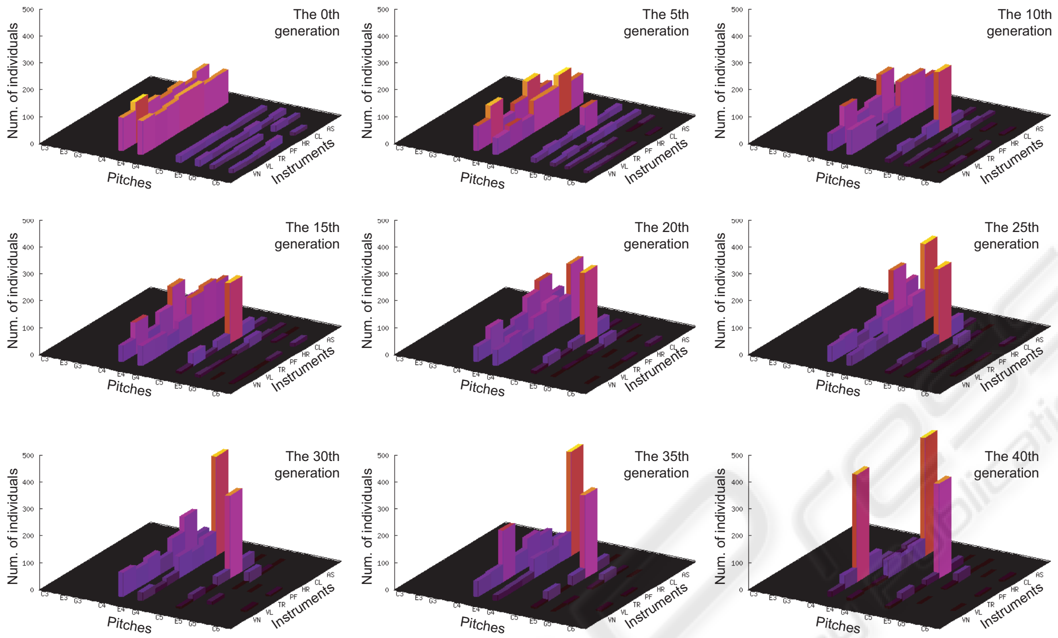

Figure 9: Population transition.

mainly driven towards finding one correct tone, then the system eventually shifts to the search of another tone. Finally, the system successfully finds a set of tones and parameters which minimizes the squared error between the target waveform and the generated one.

Next, we evaluate the robustness of the system for the increase of the number of tones and the instability of the amplitude. Figures 12 and 13 show the accuracy of the pitch and instrument estimation on each condition. We have performed 5000 evolutionary runs for every condition. We can obtain the pitch estimation accuracy of more than about 90% for two tones and 70% for three tones when the time region of the target sound begins from 250ms, 500ms, or 750ms. On the other hand, the accuracy decreases when the time region of the input sound is from 125ms. This is because a waveform of the input sound is in the *attack* or *decay* region, where the amplitude of the waveform changes significantly. For the practical application to the automatic pitch estimation, we should introduce additional information which improves the robustness of the system.

5 CONCLUSION

In this paper, we have presented a possibility of the GA-based pitch estimation method through the experimental pitch estimation of the musical consonance. An experimental pitch estimation system has a capability of analyzing the pitches of consonances, which have not been realized by the conventional methods.

REFERENCES

Back, T., Hammel, U., and Schwefel, P. H. (1997). Evolutionary computation: Comments on the history and current state. *IEEE Trans. Evolutionary Computation*, 1(1):3 – 13.

Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wasley Publishing Company.

Goto, M. (2004). Development of the rwc music database. *Proceedings of the 18th International Congress on Acoustics (ICA 2004)*, pages I–553–556.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press.

Klapuri, A. (2003). Multiple fundamental frequency estimation based on harmonicity and spectral smoothness. *IEEE Transactions on Speech and Audio Processing*, 11(6):804–816.

Piszczalski, M. and Galler, B. (1977). Automatic music transcription. *Computer music journal*, 1(4):24–31.

Pollastri, E. (2002). A pitch tracking system dedicated to process singing voice for musical retrieval. *Proceed-*

ings of 2002 IEEE International Conference on Multimedia and Expo (ICME).

Roads, C. (1985). Research in music and artifical intelligence. *ACM Computing Surveys*, 17(2):163–190.

Roads, C. (1996). The computer music tutorial. *MIT Press*.

Sterian, A. and Wakefield, G. (2000). Music transcription systems: from sound to symbol. *Proceedings of AAAI-2000 workshop on artifical intelligence and music*.

Tadokoro, Y., Matsumoto, W., and Yamaguchi, M. (2002). Pitch detection musical sounds using adaptive comb filters controlled by time delay. *Proceedings of 2002 IEEE International Conference on Multimedia and Expo (ICME)*, P03.

Tadokoro, Y., Morita, T., and Yamaguchi, M. (2003). Pitch detection of musical sounds noticing minimum output of parallel connected comb filters. *Proceedings of 2003 IEEE Region 10 Conference on Convergent Technologies For The Asia-Pacific (TENCON)*, tencon-072.

Tadokoro, Y. and Yamaguchi, M. (2001). Pitch detection of duet song using double comb filters. *Proceedings of 2001 European Conference on Circuit Theory and Design (ECCTD)*, I:57–60.

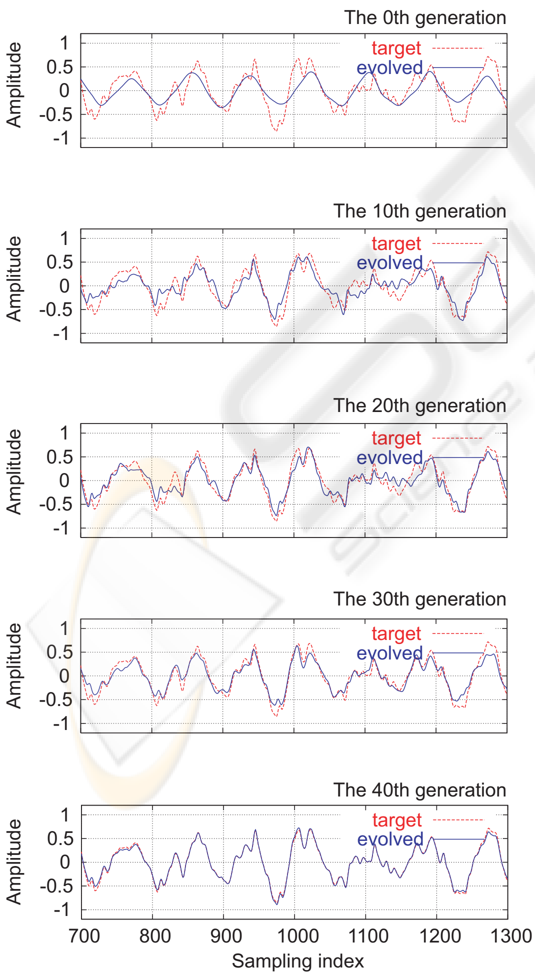

Figure 10: Waveform transition.

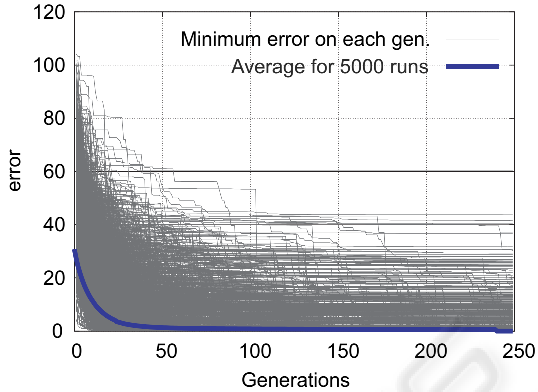

Figure 11: Transition of the sum of squared errors between the target waveform and the synthesized one.

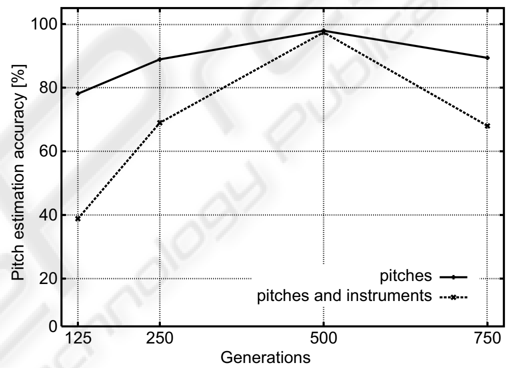

Figure 12: Accuracy of pitch/instrument estimation for two notes.

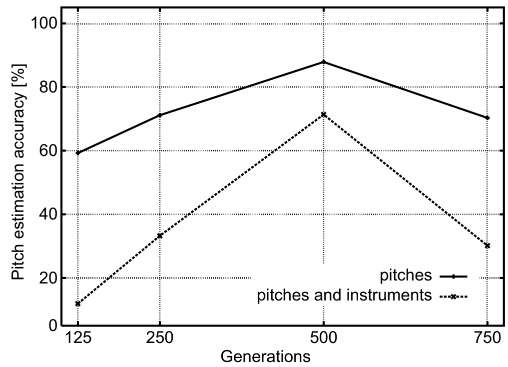

Figure 13: Accuracy of pitch/instrument estimation for three notes.