Creating Singing Vocal Expressions by means of Interactive Evolutionary Computation

Akio Watanabe, Makoto Tanji, Hitoshi Iba
IBA Laboratory, Dept. of Electrical Engineering and Information Systems,
Graduate School of Engineering, The University of Tokyo
IBA Laboratory, Dept. of Electrical Engineering, 7-3-1, Hongou, Bunkyo-ku, Tokyo 113-8656, Japan
email:akio@iba.t.u-tokyo.ac.jp, tanji@iba.t.u-tokyo.ac.jp, iba@iba.t.u-tokyo.ac.jp

***Abstract*—Today, researches for singing by computer have attracted attention. VOCALOID[1] is an application to realize that aim. By inputing lyrics and melody, users can make songs sung by the computer. In order to make the singing voice more" human", users must control frequency curve very carefully. Comparing with inputing lyrics or melody, this controlling presents heavy overhead for users. In this research, we propose a system for easily optimizing frequency curves. This system searches for parameters with a type of GA called Interactive Evolutionary Computation (IEC). On the other hand, the system using IEC has a phase for users to evaluate, we need to consider the tiredness of users. This tiredness is connected to the effectiveness of the search with GA. In this research, for the analysis of the tiredness of users, we evaluated the convergence performance of GA to fit the goal which is known in advance. As a result, we found that our method has better convergence performance than a previous method.**

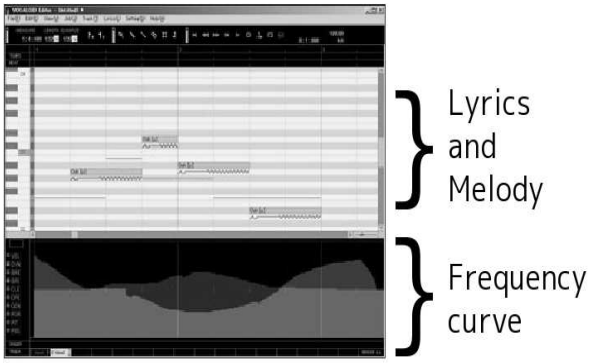


Fig. 1. Screenshot of VOCALOID

I. Introduction

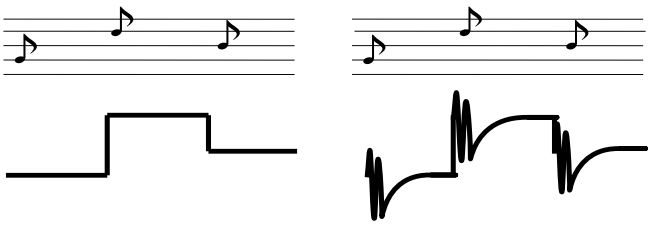
In the past, many researchers studied about music or singing voice of humans. Tanji and his colleagues proposed the method to analyze metrical structure of music by computer [12]. And with them underneath, there are also some existing researches conducted on "singing by computer". VOCALOID is such an application that realizes this [1]. In this application, users can make computer sing by inputting melody and lyrics. Although, there are many applications to support creating music [2], they cannot generate songs and lyrics. However, VOCALOID enables us to create not only musics but also songs with lyrics. So users can express more aesthetic emotions. Today, many users make various songs, and on the web, users evaluate their music with each other. This shows that VOCALOID attracts great attention.

Fig. 1 is a screenshot of VOCALOID. The top part of the screen is for inputting melody and lyrics, and the bottom of the screen is for tuning the frequency curve. In VOCALOID, we tune this curve by drawing with mouse. By tuning the frequency curve, the user can control details of singing voice, and higher quality songs can be made. But, on the other hand, since tuning needs knowledge about the singing voice, it is somewhat complicated. Many of the songs made with VOCALOID on web aren't even touched in this frequency curve. This also shows the difficulty of this tuning. Fig. 2 shows images of frequency curve. (a) is not tuned. (b) is tuned. The frequency curve of the human singing voice has some features like overshoot or vibrato [3]. If user cannot tune frequency curve and inputs only melody line on VOCALOID, song will have unnatural voice and is like (a), and listeners will easily realize this song is sung by the computer. The goal of this research is to automate this control in perspective of optimizing parameters.

There have been several previous studies for tuning such frequency curves. For example, Nakano and his colleagues developed a system called VocaListener which can tune frequency curves and make some features of singing voice like vibrato by mimicking human singing [4]. This system updates frequency curve by comparing singing of VOCALOID with that of human user. Users of this system admire it because accomplishments tuned by it have so much humanity. But this system needs a song sung by a human for mimicking, a restriction because our goal is to make computers sing like humans without human support. Therefore, we propose a method to tune frequency curves without examples of human singing.

Saito and his colleagues developed a system called SingBySpeaking that can convert speaking voice to singing voice by attaching frequency curve on speaking voice [11]. They decided the following frequency model to attach in this system.

[1]http://www.crypton.co.jp/mp/pages/prod/vocaloid/



(a) Default (b) Tuned

Fig. 2. Examples of frequency curve

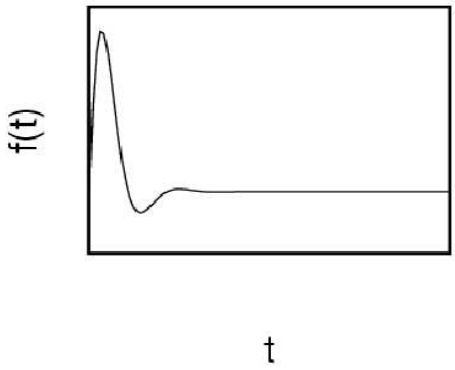


Fig. 3. An example of curve used in SingBySpeaking (for one note)

$$f(t) = \frac{k}{\sqrt{1-\zeta^2}} exp(-\zeta\omega t) sin(\sqrt{1-\zeta^2}\omega t) \quad (1)$$

(1) is the frequency model of them. Here, t is time. They decided the constant value k, ζ and ω of this model in their research, and by attaching it to each note, we can get the frequency curve which has features of the human-like singing voice like vibrato and overshoot. We can change amplitude of vibration with k, attenuation with ζ, and wavelength with ω. SingBySpeaking first gives melody to speaking voice, and then attaches this model to make human-like singing voice. The frequency curve of SingBySpeaking is like Fig. 3. And by changing the constant values, it can make vibrato too.

But, in real singing, there are various patterns of frequency curves, so the flexibility of this model is insufficient. Furthermore, this system gives the same frequency curve with the same parameters to all notes in a song, so all notes have the same way of singing. Saito and his colleagues also reported that wavelength of frequency curve for a given sing may change even though it was sung by the same person [10]. So we need delicate tuning of frequency for each note. In section IV, we show the lack of flexibility of Saito's model with some experiments.

In this paper, we try to remake the various ways of singing with computer, and find one of them which is the most favorable for user and provide it. Our another goal is to reduce the tiredness of user as much as possible. To achieve those goals, we propose a new frequency model that has more flexibility to realize various singing vocal expressions. And using this model, we implement the system to search favorable frequency curve for user. To see if we could achieve our goals, we evaluate this system by examining efficiency of searching.

The contribution of this paper is as follows:

1) Simplifying the optimization of the frequency curve without examples of human singing
2) Verifying the limit of the previous model and the effectiveness of the proposed model

The rest of this paper is structured as follows. Section II describes our new frequency model and method to optimize it. Section III explains specification of our system and the method of searching by GA. Section IV describes the method and the result of evaluation. Section V discusses our approach, followed by the conclusion in Section VI.

II. Proposed method

A. frequency model

Making the frequency curve fully automatic is a hard task for the computer, in creating singing expressions. That is because the singing voice preference differ from person to person, and there is no absolute standard for measuring them. Hence, engineers can't define and apply it to systems universally. In this research, we used a method called Interactive Evolutionary Computation (IEC), which optimizes parameters with the evaluation by a human user. With this approach, although all processes of controlling are not completed by the computer, we can give much flexibility to the frequency model, so it can produce frequency curves which each user likes. To analyze formula(1) more easily, we condensed formula(1) by changing constant value with (2)(3)(4).

$$k' = \frac{k}{\sqrt{1-\zeta^2}} \quad (2)$$

$$\zeta' = \zeta\omega \quad (3)$$

$$\omega' = \sqrt{1-\zeta^2}\omega \quad (4)$$

And the condensed formula(1) is shown in (5).

$$f(t) = k' exp(-\zeta' t) sin(\omega' t) \quad (5)$$

The envelope of sine function in formula(5) has the simple form of an exponential function, therefore this form doesn't have ability to express a flexible wave line. We decided on the new frequency model of (6).

$$f(t) = k(t) sin(\omega t + \theta) \quad (6)$$

We change the $k' exp(-\zeta' t)$ of formula(5) into $k(t)$ which is a more general function of t, and add a degree of freedom of phase in sine function.

In the system of this research searches $k(t)$ from all functions with four arithmetic operations, sine function, and exp function, and searches ω and θ as indefinite numbers. This model can express overshoot or vibrato by changing $k(t)$. So

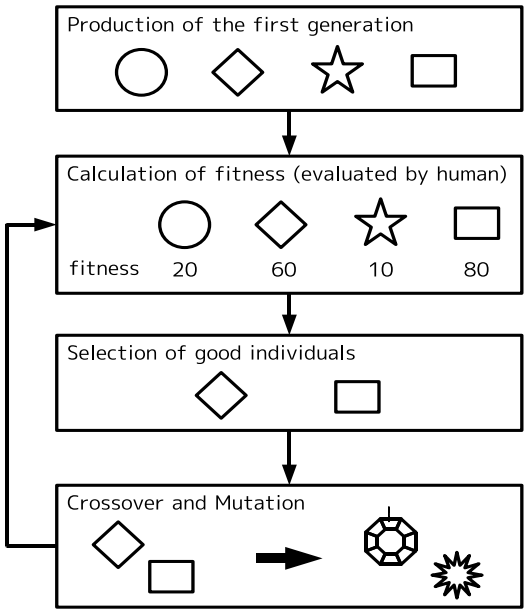


Fig. 4. Flow chart of IEC

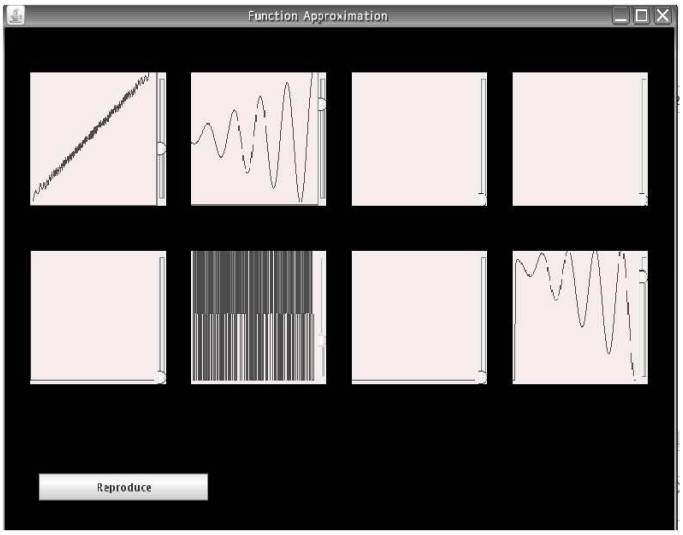


Fig. 5. Screenshot of this system

we expect this model is sufficient to meet favor of each user. But, because users must search parameters from a wider search space, we must also consider minimizing user tiredness.

B. Interactive Evolutionary Computations

1) Introduction: For the task evaluation function which can't be defined easily, IEC [5], which is a kind of Genetic Algorithm (GA), is considered in the recent years. IEC executes the part of evaluation and reproduction in the process of GA with interactions between user and computer. By this, we can introduce subjective evaluation of users to evaluation part, and the search reflects impression or favor of human. First we will explain the flow of normal GA. At first, system produces some individuals for first generation and evaluate them with the given evaluation function. Then it chooses parents from good individuals and these parents make crossover or mutation to create next generation. And system evaluates this next generation. By repeating reproduction and evaluation, system can get better and better individuals. In the normal GA process, there is a certain evaluation function which is defined by engineers, and evaluation result of each individual is given to system with the form of some value. So the computer can select good individuals without other information. Because the defined evaluation function is needed, normal GA systems can't solve the task whose evaluation function is impossible to be defined such as composition or art design. But in IEC systems, we don't need to define the evaluation function. Fig. 4 is a flow chart of IEC. In IEC systems, users directly evaluate individuals which are products of computer, and computer reproduces better (at least for the user) individuals and user re-evaluates these individuals. By repeating this process, computer can produce high-quality work for user. Put simply, it is a way of computation which uses human feelings instead of the evaluation function in normal GA.

There are many applications using IEC [6]. Ando and his colleagues developed a system called CACIE which can make musics using IEC [9]. All users have to do is to evaluate songs. This system make composition much easier than before.

2) Features of IEC: As we mentioned in the previous section, first we can say one feature of this IEC system is that the system doesn't need an evaluation function. Therefore, the system designer can make the system that can make works of high satisfaction level for user without knowledge about the works. In the case of songs, Engineer doesn't have to define what song is good or bad.

Evaluations consume a lot of time and users can easily get tired during the process, which are disadvantages of the approach. Unlike problems which can be completely solved only by the computer, IEC puts a strain on users for searching, so we should reduce the tiredness of users as much as possible.

III. Details of the System

We implemented automatic optimizing frequency curve system using IEC with the model explained in section III. The goal of this system is to optimize the frequency curve more easily and without good examples.

A. Specification

First, the system makes 8 individuals with random parameters, and then system makes 8 frequency curves with parameters of each individual and converts it into the data files of VOCALOID. Next, the VOCALOID system makes wave files from tuned data files. Afterwards, user listens to the wave files and evaluates them. At the time of evaluation, user can see frequency curves visually in the interface of this system.

Fig. 5 is a screenshot of GUI. 8 individuals of GA are reflected to the frequency curve of VOCALOID and frequency curves are drawn on the GUI. User listens to the music which

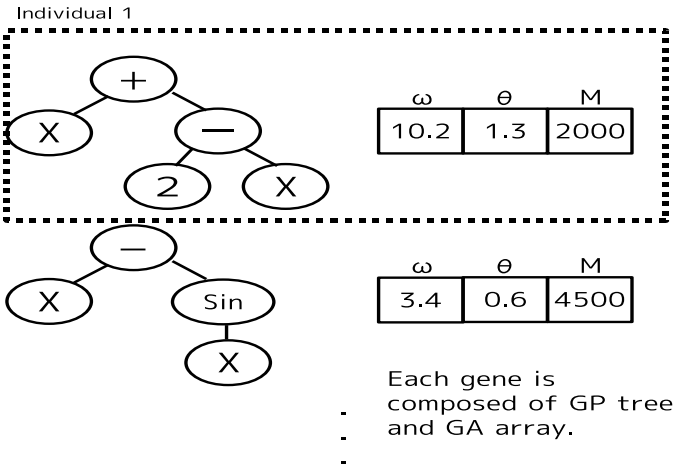


Fig. 6. Composition of each gene

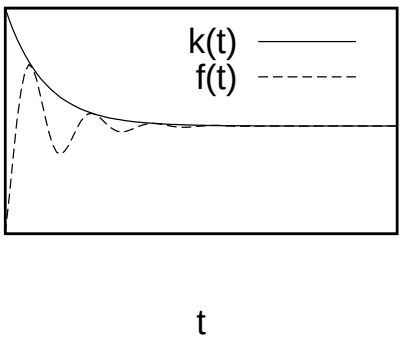


Fig. 7. Example of k(t) in our model

8 individuals have, and evaluates each individual with bars at the right side of each curve graphic. This evaluation is used in IEC system, and the highly evaluated individuals are selected as parents to generate the next generation. Then crossovers and mutation are executed. And in this process, parameters of frequency curve are optimized and the user can get the frequency curve which has favorable singing vocal expression.

B. Searching for parameters of model

Each gene of GA in this research has one tree structure and one array which has 3 values. Fig. 6 is an image of gene. The tree structure of gene gives envelope of frequency line like $k(t)$ of Fig. 7. We expressed $k(t)$ with tree structure, and M, ω and θ with values in array. Next we will explain more precisely about gene. $k(t)$ of our model is constructed with Genetic Programming (GP) [7] which is a kind of GA and has a tree structure.

Example of a GP tree is shown at left side of individual 1 in Fig. 6. In this example, x and $(2-x)$ hook on plus mark, so this entire tree structure means $x+(2-x)$. Crossover of GP is done by changing a part of tree with other trees. Mutation of GP is done by changing a part of a tree randomly. Parameters of GP in this research are shown in Table I.

TABLE I
GP PARAMETER

Population size	8
Selection	tournament
Tournament size	2
Crossover rate	0.8
Mutation rate	0.2
Elite size	1
Max depth	6
Min depth	3

TABLE II
GA PARAMETER

Population size	8
Selection	tournament
Mutation rate	0.1
Elite size	1

We used x, four arithmetic operations, sin and exp as nodes. To optimize ω which deals with wavelength and θ which deals with phase, we used GA. In GA, we use UNDX [8] to implement crossover. In UNDX, crossover makes child with probabilistic distribution of Gaussian whose average is the medium value of 2 parents, and whose dispersion equals the half of distance between 2 parents.

Parameters of GA in this research are shown in Table II.

In the case of making function with GP, the maximum value often becomes too large, so we must execute scaling for GP. And the average of $f(t)$ should be zero as not to change melody line of original song. So we made the following operation for functions.

(1)System calculates the average and subtracts it from the entire function. Then the average of $f(t)$ becomes zero.

(2)System finds a value in the function whose absolute value is the highest, and to make it become M, system multiplies the entire function by a constant value. M is also optimized with GA.

IV. DELIBERATION FOR THE TIREDNESS OF USER

A. The way of evaluation for this system

Average users give up manual tuning of frequency curve because of tiredness. One of the main goals for this system is to reduce the tiredness of user compared with the manual tuning of parameters, so we need to evaluate the tiredness which is needed to optimize parameters using this system. User's tiredness is connected to the efficiency of search of evolutionary computation. In this research, we executed automatic searching for the given frequency curve, and examine the relation between number of generations and correctness of remade curve. For the evaluation, we remade the frequency curve of each note of a part of the song "AKATONBO". After making frequency curve with the previous model and the proposed model, we calculated mean absolute error between made curve and the real curve of each note sung by real

TABLE III
MEAN ABSOLUTE ERROR OF EACH MODEL (BEST)

note	Previous model	Proposed model	SimpleGP model
yu	1318.51	903.79	880.77
u	2277.34	2274.61	1993.25
ya1	1262.02	1015.82	1201.76
ke1	1078.76	722.37	710.62
ko	2436.73	2369.60	2374.51
ya2	1682.71	792.20	940.62
ke2	573.07	350.07	363.46
e	354.66	338.15	227.04
no	1661.40	1460.54	1554.43
a	1440.70	1406.04	1406.04
ka	1596.59	1426.52	1447.44
to	1045.28	888.20	1186.11
n	561.18	421.78	459.01
total	17288.93	14369.69	14745.07

TABLE IV
MEAN ABSOLUTE ERROR OF EACH MODEL (AVERAGE)

note	Previous model	Proposed model	SimpleGP model
yu	1768.98	1074.77	1044.23
u	2292.67	2274.61	2105.80
ya1	1325.82	1072.85	1211.38
ke1	1142.59	790.22	892.31
ko	2515.31	2402.09	2390.54
ya2	1682.71	970.3	1168.27
ke2	699.28	401.9	392.78
e	473.28	339.14	294.32
no	1837.45	1471.67	1556.92
a	1535.01	1406.04	1442.11
ka	1816.62	1520.68	1544.84
to	1257.84	1023.26	1216.41
n	614.39	449.45	485.26
total	18961.94	15196.96	15745.17

human. And we examined how many generations are needed to get closer to the curve which has enough similarity to the curve of human singing. In the case of using IEC, user's number of times of evaluation is the product of the number of individuals per generations and the number of generations, so with this experiment, user's tiredness for using this system can be examined. And for comparison, we also executed the search using a simple GP model which gives all functions composed of four arithmetic operations , exp function and sine function.

B. The search process of the previous model

For the previous model, we searched k, ζ and ω using GA. Parameters of GA are the same as in Table II. In Saitos' process they applied the same parameters to all notes, but in this experiment we searched parameters optimized for each note.

C. Result

Table III shows the absolute error between the best curve in 50th generation and the real curve. A lower value means higher accuracy of remaking real singing voice. All shown values are the bests of 10 trials.

The note "ya2" shows a remarkable difference between the model in Table III. And Fig. 8 shows transitional change of the mean absolute error of the trial which makes best individual for "ya2". Vertical axis shows mean absolute error, so the lower is the better.

TABLE V
MEAN ABSOLUTE ERROR OF EACH MODEL (STANDARD DEVIATION)

note	Previous model	Proposed model	SimpleGP model
yu	225.19	147.98	114.93
u	48.42	0.00	145.29
ya1	69.60	68.29	20.46
ke1	201.82	73.78	71.89
ko	31.61	29.08	17.55
ya2	0.01	165.21	144.01
ke2	399.01	86.04	31.32
e	251.18	1.96	59.70
no	61.86	20.47	3.57
a	106.25	0.00	38.88
ka	123.2	97.59	89.45
to	154.03	102.44	36.33
n	118.60	26.81	24.51

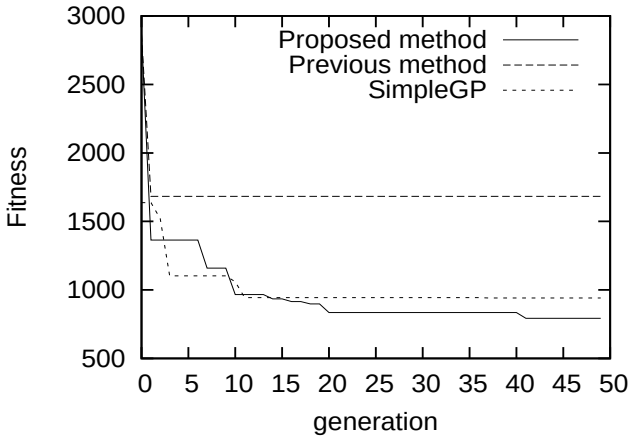


Fig. 8. Fitness transition of each model (ya2)

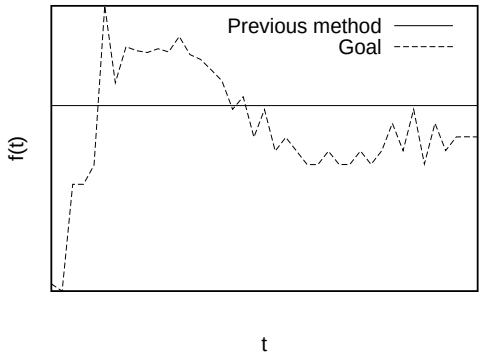


Fig. 9. Best individual of previous model

Figs. 9-11 are the best individuals of each model for remaking "ya2".

V. DISCUSSION

By looking at Table III, it is evident that the results of all notes of the proposed model are better than those of the

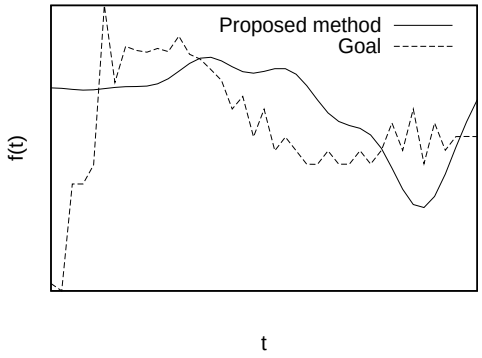


Fig. 10. Best individual of proposed model

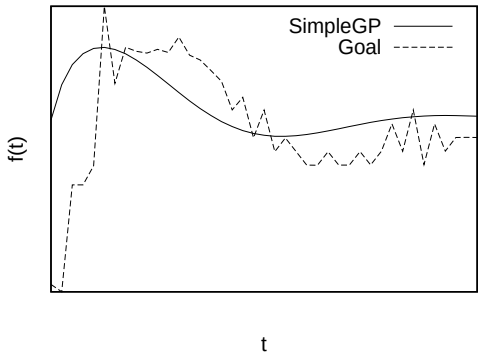


Fig. 11. Best individual of simple GP model

previous model. The same is true of the averages shown in Table IV. From this, we can verify the effectiveness of the flexibility of the frequency model. Table V shows standard deviations. Since these values are small, we can say 50 generations are enough for convergence.

We can also see the effectiveness of this model in Fig. 8. The mean absolute error of the previous model converges by 5th generation and it doesn't become better thereafter. But the mean absolute error of the proposed model can be better after the 5th generation because of its flexibility, and we can see the mean absolute errors of the two models differed greatly at 50th generation. And the model of simple GP has a slightly worse result than the proposed model, but it doesn't have much difference. In this research, we try to find a model that is more suitable for search than a simple GP model which accomplishes searching more quickly, but we think there is room for improvement in the model.

From Figs. 9-11, we can check the limit of searching with the previous model. The frequency curve of this note can't be expressed with the previous model correctly and the searching ends regarding $f(t) = 0$ as the best individual. Comparing this, the proposed model and the simple GP model can follow the change of the curve, though it did not perfectly remake the curve.

VI. Conclustion

We implemented a system which optimizes the frequency curve automatically using IEC for reducing the tiredness of users of VOCALOID. For the evaluation of user's tiredness, to measure how many times users need to evaluate individuals to complete optimization, we remake the frequency curve of songs sung by human with the system, and examine the relationship between the number of times of evaluation and the quality of remaking. As a result, we found the proposed model can express singing voice more correctly than the previous model, and in aspects of efficiency of search, the proposed model shows a slightly better result than the simple GP model. This is not enough for the aim of our system to complete optimization by about 20 generations. But we can expect that we can achieve our aim by constructing more accurate frequency model.

And we need to evaluate user's tiredness more directly. When users use this system as a practical matter, they don't have some examples for imitation, and what they do is to choose individuals which they like for executing evolution. Therefore, there is a gap between evaluation in this research and in practice. The results prove that the proposed model quickly achieves convergence compared to the previous model. And we cannot say how many generations are needed to achieve convergence in practice; we should send out questionnaires about tiredness or user impression.

VII. acknowledgement

I have had the support and encouragement of Dr. Daichi Ando.

References

[1] H Kenmochi, H Ohshita, "Singing synthesis system 'VOCALOID'", 2007-MUS-72-(5), pp. 25-28
[2] "band in a box", http://www.cameo.co.jp/PG/win/
[3] J Sundberg,"Research on the singing voice in retrospect" *TMH-QPSR*, vol.45-1, 2003, pp. 11-22
[4] T Nakano, M Goto,"VocaListener: An Automatic Parameter Estimation System for Singing Synthesis by Mimicking User's Singing", 2008-MUS-75, No.50, pp. 49-56
[5] H Takagi, T Unemi, T Terano, "Perspective on Interactive Evolutionary Computing", *Proceedings of the Annual Conference of JSAI*, September,1998
[6] M Sugawara, M Mitsunori, T Hiroyasu, "Interactive genetic algorithm using generates initial individual based on a favorite color image", JSAI 2008, 2B1-2
[7] J R Koza, R Poll,"GENETIC PROGRAMMING"
[8] I Ono, M Yamamura, H Kita, "Real-Coded Genetic Algorithms and Their Applications" *Proceedings of the Annual Conference of JSAI*, 2000
[9] D Ando, P Dahlsted, M Nordahl, H Iba, "CACIE: Computer Aided Composition System by Interactive Evolutionary Computation", 2005-MUS-59-(10), pp. 55-60
[10] T Saitou, M Unoki, M Akagi, "A study on a control method of vibrato modulation frequency for synthesizing natural singing-voice",*IEICE technical report*, 2005
[11] T Saitou, M Goto, M Unoki, M Akagi, "SingBySpeaking: Singing Voice Conversion System from Speaking Voice By Controlling Acoustic Features Affecting Singing Voice Perception" *IPSJ SIG Notes 2008(12)* pp.25-32, 2008
[12] M Tanji, D Ando, H Iba, " Improving Metrical Grammar with Grammar Expansion" *AI 2008: Advances in Artificial Intelligence* pp. 180-191, 2008