# Mandarin Singing Voice Synthesis Using an HNM Based Scheme

Hung-Yan Gu and Huang-Liang Liau
*Dept. CSIE, National Taiwan University of Science and Technology, Taipei, Taiwan*
*E-mail: guhy@mail.ntust.edu.tw*

## Abstract

*In this paper, HNM (harmonic plus noise model) is enhanced and used to design a scheme for synthesizing Mandarin singing voice. Enhancements made include synthesizing signals with higher fluency level and keeping the timbre of synthetic singing voices consistent. In terms of the signal synthesis equations rewritten here, a Mandarin singing voice synthesis system is constructed, in which each Mandarin syllable is recorded only once. This system can parse a song score file and synthesize its lyric syllables' signals in real-time. Besides, the skill of portamento singing is realized. According to perceiving the synthetic songs, the timbre is indeed consistent, and the signals are very clear and natural.*

## 1. Introduction

Several techniques were proposed for a computer to synthesize music [1, 2], including additive synthesis, subtractive synthesis, and FM (Frequency Modulation) synthesis. In this paper, however, we choose to enhance the technique of HNM (harmonic plus noise model) and use it to design a scheme for synthesizing Mandarin singing voice in order to obtain high signal clarity and naturalness level. HNM is originally proposed by Y. Stylianou [3, 4] and is thought to belong to the class of additive synthesis. It splits the spectrum of a signal frame into two halves of unequal widths in order to better model the spectrum. The lower frequency half is modeled as consisting of harmonic partials while the higher frequency half is modeled as consisting of noise signal components.

Mandarin is a syllable prominent language, and each syllable is of the structure, $C_xVC_n$. The initial, $C_x$, may be null, a voiced consonant, or an unvoiced consonant while the final, $C_n$, may be null or a nasal as /n/ or /ng/. As to the nucleus, V, it may be a vowel, diphthong, or triphthong. Therefore, we take syllable as the unit for singing voice synthesis.

In this paper, the HNM based scheme is as depicted in Fig. 1. First, a note's data is inputted and parsed.

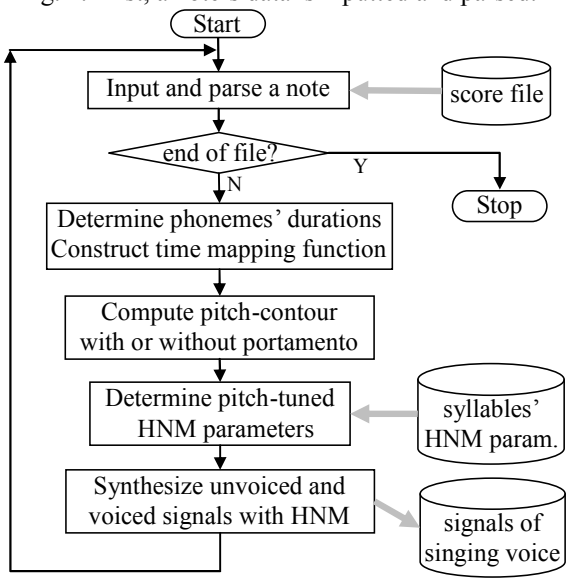


**Figure 1. Main flow of the HNM based scheme.**

Then, the durations of the comprising phonemes are determined for the note's lyric syllable. In terms of these durations, a time axis mapping function can be constructed. More detailed explanation is in Section 3.1. Next, a pitch contour for the note is computed. If the lyric syllable is sung in a portamento, the computation of pitch contour becomes more complicated. This is explained in Section 3.5. To keep timbre consistent, the HNM parameters of the lyric syllable must be adjusted in a correct way. This is explained in Section 3.2. In the last block of Figure 1, the signals for the unvoiced and voiced parts of the lyric syllable are synthesized with HNM.

If the $C_x$ part of a syllable is a short unvoiced consonant (e.g. /b, d/), its synthetic signal will be directly copied from the corresponding part in the recorded syllable. If the $C_x$ part is a long unvoiced consonant (e.g. /s, p/), its synthetic signal will be generated as noise signal with HNM. Otherwise, the $C_x$

is a voiced consonant (e.g. /m, r/) and will be considered together with the remaining phonemes. Because the remaining phonemes of a syllable are all voiced, their synthetic signal will be generated as harmonic partials plus noise signal with HNM. The methods for synthesizing harmonic and noise signals are explained in Section 3.3 and 3.4.

## 2. Score file parsing

In a song score file, each line except the first line contains one note's information, i.e., pitch symbol, number of beats, lyric. The information contained in the first line are song name, tempo (e.g., 120 means 120 beats per minute), and duty ratio (e.g., 85 means 85% of a note's duration is used in singing). Parsing is to slice out the 3 fields in a line and to interpret the meanings of these fields.

The pitch symbol of a note is of the format "XYZ" (e.g. G3#). "X" denotes the tone name, "Y" denotes the tone range, and "Z" is "#" (sharp) or "b" (flat). Through interpretation, the pitch symbol is converted to a numeric value of pitch frequency. After all notes' pitch frequencies are determined, automatic key shifting is executed. Key shifting must be done in order to translate the pitch range of the score file to match the pitch range of the person who utters the Mandarin syllables as the synthesis units.

Here, automatic key shifting is done in the following steps: (a) Find the maximum and minimum values from the notes' pitch frequencies; (b) Take the average of the maximum and minimum values found; (c) Compute the ratio of the person's analyzed mean pitch frequency to the average value; (d) Multiply each note's pitch frequency with the ratio computed.

After the number of beats for a note is parsed, the tempo value in the first line can be used to compute the time-length of a note. However, a note is usually not sung in its full length because some small ratio of this length is reserved for breathing or transiting to its following note.

Next, the lyric of a note is parsed. Usually, each note has a unique lyric syllable assigned to it. But sometimes two or three consecutive notes may be assigned a same lyric syllable (i.e. portamento). This situation is indicated in the score file with a convention. When a note is to be assigned the same lyric as its preceding note, the third field for this note will be placed a special character such as "|".

## 3. Signal waveform synthesis

How to keep the timbre of synthetic syllables consistent? The implementation method is not given in the original HNM [3, 4]. Note that each Mandarin syllable has only one recorded utterance here. When the pitch frequency of a syllable is changed, the values of the syllable's HNM parameters should be adjusted in a way that the timbre can be kept consistent. Also, how to warp the time axis of a synthetic syllable in order that more fluent syllable signal can be synthesized? The solution to this issue is not found in the original HNM. Note that a syllable's duration needs to be lengthened or shortened, and a linear time warping way usually results in lower perceived fluency.

### 3.1. Planning of phoneme duration

When a syllable is started with a short-unvoiced phoneme, e.g. /bau/, the time length of the short-unvoiced is planned as the corresponding phoneme length in the recorded syllable. But when started with a long-unvoiced phoneme, the length of the long-unvoiced is planned by multiplying its original length with a factor *Fu*. *Fu* is computed as the synthetic syllable's length divided by the recorded syllable's length. But its value is confined to within the range from 0.6 to 1.4.

Consider the example syllable, /man/. Suppose in the recorded signal of /man/, the three phonemes, /m/, /a/, and /n/, occupy *Rm*, *Ra*, and *Rn* ms, respectively, and $Rv = Rm + Ra + Rn$. Also, suppose that *Dm*, *Da*, and *Dn* represent the time lengths of the three phonemes within the synthetic syllable, and $Dv = Dm + Da + Dn$. Then, we plan the values of *Dm*, *Da*, and *Dn* with a procedure that iteratively decreases the values of *Dm* and *Dn*, and increases the value of *Da* till the ratio, $Da / Dv$, is greater than a defined value (e.g. 0.5).

After the values of *Dm*, *Da*, and *Dn* are determined, a mapping function from the phonemes in the synthetic syllable to the corresponding phonemes in the recorded syllable can then be established. This mapping function is as depicted in Figure 2, i.e. a piece-wise linear time

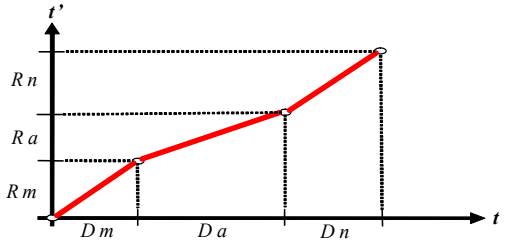


**Figure 2. Piece-wise linear mapping function.**

warping function. According to the constructed mapping function, an analysis frame's time position on a recorded syllable's time axis can then be mapped to a time point on a synthetic syllable's time axis. A

mapped time point is also called a control point [1, 2].

### 3.2. Pitch-tuned HNM parameters

On a control point, the pitch-original HNM parameters, $A_i$ (amplitude), $F_i$ (frequency), and $\theta_i$ (phase), for the $i$-th harmonic partial can be obtained by referring to its corresponding analysis frame. However, the parameters, $\tilde{A}_k$, $\tilde{F}_k$, and $\tilde{\theta}_k$, for the $k$-th pitch-tuned harmonic partial should be determined carefully in order to keep timbre consistent. To have consistent timbre, a principle is to keep the spectral envelope unchanged [2]. This implies that the amplitude $\tilde{A}_k$ of the $k$-th pitch-tuned harmonic partial located at frequency $\tilde{F}_k$ must be determined according to the spectral envelope defined by the sequence of pairs, ($F_i$, $A_i$). For example, in Figure 3, the

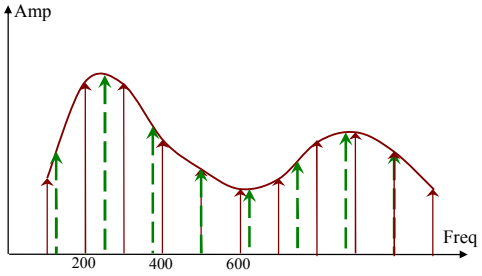


**Figure 3. Keep spectral envelope while tuning pitch.**

amplitudes of the pitch-original harmonic partials represented by solid lines define the spectral envelope curve. According to this curve, the amplitudes of the pitch-tuned harmonic partials represented with dashed lines are determined. In details, for the $k$-th harmonic frequency $\tilde{F}_k$, we first find a pitch-original harmonic frequency, $F_j$, from $F_1$, $F_2$, $F_3$, …, that is nearest to and less than $\tilde{F}_k$. Then, the four pitch-original partials of the frequencies, $F_{j-1}$, $F_j$, $F_{j+1}$, and $F_{j+2}$, are used to perform order three Lagrange interpolation to compute the value of $\tilde{A}_k$. That is,

$$\tilde{A}_k = \sum_{m=j-1}^{j+2} A_m \cdot \prod_{\substack{h=j-1 \\ h\neq m}}^{j+2} \frac{\tilde{F}_k - F_h}{F_m - F_h} \qquad (1)$$

Similarly, the phase $\tilde{\theta}_k$ of the pitch-tuned harmonic partial located at frequency $\tilde{F}_k$ can be interpolated with the four pitch-original partials of the frequencies, $F_{j-1}$, $F_j$, $F_{j+1}$, and $F_{j+2}$. However, the phases of the four partials, $\theta_{j-1}$, $\theta_j$, $\theta_{j+1}$, and $\theta_{j+2}$, must be unwrapped beforehand to prevent phase discontinuities. That is, the unwrapped phases, $\hat{\theta}_{j-1} = \theta_{j-1}$, $\hat{\theta}_j = puw(\theta_j, \hat{\theta}_{j-1})$, $\hat{\theta}_{j+1} = puw(\theta_{j+1}, \hat{\theta}_j)$, and $\hat{\theta}_{j+2} = puw(\theta_{j+2}, \hat{\theta}_{j+1})$, are used instead in the interpolation processing. The phase unwrapping function above is defined here as

$$puw(x, y) = x - M \cdot 2\pi \qquad (2)$$

$$M = \left\lfloor \frac{1}{2\pi}(x - y + \theta) \right\rfloor, \quad \theta = \begin{cases} \pi, & \text{if } x \geq y \\ -\pi, & \text{otherwise} \end{cases}$$

### 3.3. Synthesis of harmonic signal

For the harmonic signal, $H(t)$, between the n-th and (n+1)-th control points, its sample values are computed with the rewritten equations,

$$H(t) = \sum_{k=0}^{L} a_k^n(t) \cos\left(\phi_k^n(t)\right), \quad t = 0, 1, \ldots, T^n, \qquad (3)$$

$$a_k^n(t) = \tilde{A}_k^n + \frac{t}{T^n}(\tilde{A}_k^{n+1} - \tilde{A}_k^n), \qquad (4)$$

$$\phi_k^n(t) = \phi_k^n(t-1) + 2\pi f_k^n(t) / 22{,}050, \quad \phi_k^n(0) = \hat{\theta}_k^n, \qquad (5)$$

$$f_k^n(t) = \tilde{F}_k^n + \frac{t}{T^n}(\tilde{F}_k^{n+1} - \tilde{F}_k^n), \qquad (6)$$

where $L$ is the number of harmonic partials, $T^n$ is the number of samples between the $n$-th and $(n+1)$-th control points, 22,050 is the sampling rate, $a_k^n(t)$ is the time-varying amplitude of the $k$-th partial at time $t$, $\phi_k^n(t)$ is the cumulated phase for the $k$-th partial, $f_k^n(t)$ is the time-varying frequency for the $k$-th partial, and $\hat{\theta}_k^n = puw(\tilde{\theta}_k^n, \hat{\theta}_k^{n-1})$, i.e. unwrapped phase of $\tilde{\theta}_k^n$ versus $\hat{\theta}_k^{n-1}$. In Equations (4) and (6), linear interpolation is used.

Note that when using Equation (3) to synthesize signal samples, the cumulated phase, $\phi_k^n(t)$, is generally not continued at the boundary time points, i.e. $t$=0 or $t$= $T^n$. This kind of discontinuities, i.e. $\phi_k^n(T^n) \neq \phi_k^{n+1}(0)$, will induce amplitude discontinuities to signal waveform, and cause clicks to be heard. To prevent this kind of discontinuities, the amount of mismatched phase, $\xi_k^n$, at the boundary point, $t = T^n$, must be computed beforehand. Then, this amount can be divided and shared to the $T^n$ sample points between two adjacent control points. Accordingly, the phases of the signal samples around the boundary point will move smoothly. Here, the amount of mismatched phase is computed as

$$\xi_k^n = puw\left(\phi_k^n(T^n), \phi_k^{n+1}(0)\right) - \phi_k^{n+1}(0) \qquad (7)$$

where the phase unwrapping function, puw( x, y), is as defined in Equation (4) and $\phi_k^n(T^n)$ can be directly computed as

$$\phi_k^n(T^n) = \phi_k^n(0) + \frac{\pi}{22{,}050}\left((T^n + 1) \cdot \tilde{F}_k^{n+1} + (T^n - 1) \cdot \tilde{F}_k^n\right) \qquad (8)$$

The formula of Equation (8) is obtained by iteratively evaluating Equation (5) and (6). Then, by dividing and sharing $\xi_k^n$ to the samples between two control points, Equation (6) can thus be modified to

$$H'(t)=\sum_{k=0}^{L} a_k^n(t)\cos\left(\phi_k^n(t)-\frac{t}{T^n}\cdot\xi_k^n\right),\ t=0,1,...,T^n-1, \quad (9)$$

### 3.4. Synthesis of noise signal

For the noise signal, we decide to synthesize it as a summation of sinusoidal components [3]. Let $G_k$ be the frequency of the $k$-th sinusoid. Because $G_k$ do not change with time, we define $G_k = 100 \cdot k$ (Hz). However, for the $n$-th control point, the index $k$ of $G_k$ is not started from 1 and its starting value, $K_S^n$, is determined according to the MVF (maximum voiced frequency) of this control point, i.e. $K_S^n = \lceil \text{MVF(n)} / 100 \rceil$. But the end value of the index $k$ is always a fixed value, $K_e = \lfloor 11{,}025 / 100 \rfloor$.

Let $B_k^n$ be the noise amplitude for the $k$-th sinusoid on the $n$-th control point. To determine its value, the 10 cepstrum coefficients, on the $n$-th control point, representing the noise spectral envelope are first appended with zero values and inversely transformed (inverse discrete Fourier transform) to the spectral domain [3, 5]. Then, exponentiation is taken to obtain the corresponding spectral magnitude coefficients, $X_j$, $j$ = 0, 1, …, 2047. According to the magnitudes $X_j$, the value of $B_k^n$ can be obtained by linearly interpolating the two adjacent magnitudes, $X_i$ and $X_{i+1}$, whose frequencies surround the frequency of $G_k$.

When the values of $K_S^n$ and $B_k^n$ for the $n$-th control point are known, the noise-signal samples between the $n$-th and $(n+1)$-th control points can then be computed with the rewritten equations,

$$N(t)=\sum_{k=K_s}^{K_e} b_k^n(t)\cos\left(\gamma_k^n + t\cdot 2\pi G_k/22{,}050\right), \quad (10)$$

$$t=0,1,...,T^n-1,$$

$$b_k^n(t)=B_k^n+\frac{t}{T^n}(B_k^{n+1}-B_k^n), \quad (11)$$

$$\gamma_k^n=\gamma_k^{n-1}+T^n\cdot 2\pi G_k/22{,}050, \quad (12)$$

where $K_S$ is set to the lesser of $K_S^n$ and $K_S^{n+1}$, and $\gamma_k^n$ is the initial phase for the $k$-th sinusoid on the $n$-th control point.

### 3.5. Synthesis of portamento singing

Usually a lyric syllable is assigned one musical note. But occasionally a syllable may be assigned two (or three) notes. When a syllable is assigned more than one note, it should be sung in a portamento manner. That is, the pitch-contour of the syllable should be smoothly transited from the former note's pitch to the latter note's pitch in the middle portion. An example pitch-contour is shown in Figure 4. The duration of the syllable is divided into three time intervals. The left and right intervals are planned to sing stable pitches of the two notes in order that they can be explicitly perceived. And the middle interval is used to transit the pitch smoothly.

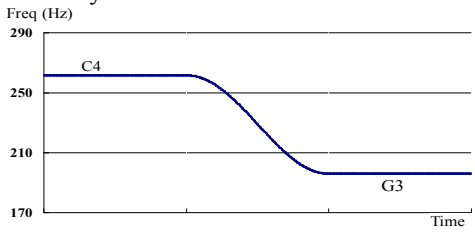


**Figure 4. Example pitch-contour in a portamento.**

In this paper, the pitch-contour of a lyric syllable is planed before its pitch-tuned HNM parameters are determined. Suppose that the two notes to be sung in portamento are of the pitch frequencies *Pa* and *Pb*. We first divide the control points within the vowel part of the syllable into three groups. Then, the control points within the first and third groups are directly assigned the pitches of *Pa* and *Pb* respectively. But for the $n$-th control point in the second group, its pitch, $P^n$, is defined with a cosine based function. That is,

$$P^n=\frac{(P_a+P_b)}{2}+\frac{(P_a-P_b)}{2}cos\left(\frac{n}{M}\pi\right) \quad (13)$$

where M is the number of control points in the second group.

## 4. System implementation and experiments

Mandarin has only 408 different syllables if superimposed tones are not distinguished. Hence, we decide to record and save each of these syllables once for analyzing HNM parameters. Each of these syllables is isolatedly uttered in level tone by a female in a sound proof room. Then, an HNM analysis program is developed to process these syllables. The analysis method used is based on the one proposed by Stylianou [3] but some modifications are made. For example, the frequency values of harmonic peaks in a spectrum are more precisely estimated with parabolic interpolation. And an analysis frame's MVF is more strictly defined as that its following five harmonic candidates are check to be not harmonic peaks.

In developing the program for synthesizing Mandarin singing voice, the methods described in

Section 2 are used to parse an input score file, and the methods described in Section 3 are used to synthesize the signal waveforms for the lyric syllables. Since the amount of computations is large, the synthesis program is hardly to run in real-time with an ordinary personal computer (e.g., a 2.6MHz Pentium CPU based). However, we intend to smoothly synthesize singing voices and play the signal waveforms in real-time. This is because our synthesis program will be integrated into a humanoid robot to show the skill of singing. Therefore, we had tried to find possible bottlenecks. As a result, a major bottleneck is found to be the inverse FFT operation for transforming cepstrum parameters back to spectrum domain for determining noise signal's amplitudes. When the inverse FFT length is changed from 4,096 to 1,024 points, the synthesis speed is largely improved and achieve 3 times of real-time. The frequency spacing of 21.53Hz (22,050 / 1,024) between two adjacent bins is thought to be sufficient because the frequencies of adjacent sinusoidal components are 100Hz apart.

To show the ability of the HNM-based synthesis scheme, spectrograms for the signals of the syllable /wan/ are analyzed with the package, WaveSurfer, and shown in the lower part of Figure 5. The spectrogram

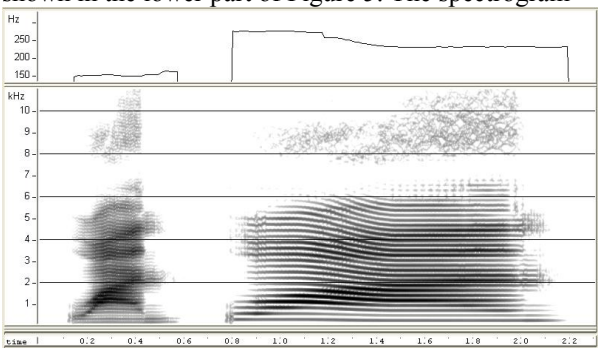


**Figure 5. Pitch-contours and spectrograms for the recorded and synthetic syllables, /wan/.**

at the left side of Figure 5 is for the recorded syllable /wan/ while the spectrogram at the right side is for a synthetic syllable /wan/ sung in a portamento manner. When the two spectrograms at the two sides are compared, it can be found that the formant traces have same curve shape and same frequency height. This explains why they will have same timbre. As seen in the upper part of Figure 5, the pitch height and shape of the synthetic syllable are very different from those of the recorded syllable. However, the clarity and naturalness of the synthetic singing signals are kept in a high level. For demonstration, we have set up a web page, http://guhy.csie.ntust.edu.tw/trhnm/sing.html. From this web page, the signal waveforms for the two syllables in Figure 5 can be downloaded and listened. In addition, some synthetic Mandarin songs are provided, which are intended to show the timbre consistency, signal clarity and naturalness as mentioned above.

## 5. Concluding remarks

In this paper, a piece-wise linear function is used to map an analysis frame of a recorded syllable to a control point of a synthetic syllable in order to promote the fluency of the synthetic singing syllables. And an order three Lagrange interpolation based method is proposed to determine the pitch-tuned HNM parameters in order to keep the timbre of the synthetic singing voice consistent. In terms of such enhancements and the signal-synthesis equations rewritten here, we have constructed a Mandarin singing voice synthesis system, in which each Mandarin syllable is only recorded once. Furthermore, by eliminating the computational bottleneck in computing noise spectrum, the system can now work smoothly in real-time. According to perceiving the synthetic songs as demonstrated in our web page, we conclude that the HNM based scheme proposed here can indeed be used to synthesize Mandarin singing voice with consistent timbre and high signal clarity.

## 6. References

[1] F. R. Moore, *Elements of Computer Music*, Prentice-Hall, 1990.

[2] C. Dodge and T. A. Jerse, *Computer Music: Synthesis, Composition, and Performance*, second edition, Schirmer Books, New York, 1997.

[3] Y. Stylianou, *Harmonic plus Noise Models for Speech, Combined with Statistical Methods, for Speech and Speaker Modification*, Ph.D. thesis, Ecole Nationale Supèrieure des Télécommunications, Paris, France, 1996.

[4] Y. Stylianou, "Modeling Speech Based on Harmonic Plus Noise Models", *Nonlinear Speech Modeling and Applications*, Springer-Verlag, Germany, 2005.

[5] D. O'Shaughnessy, *Speech Communications: Human and Machine*, second edition, IEEE Press, 2000.

## Acknowledgement

The authors would like to acknowledge the financial support from National Science Council of Taiwan under Grant No. NSC-95-2218-E-011-009.