Transcription to Instrumental Music Synthesis with Tabla

Uttam Kumar Roy
Dept. of Information Technology,
Jadavpur University,
Kolkata, India
u_roy@it.jusl.ac.in

***Abstract*—The *Tabla* is one of the most sophisticated percussion instruments extensively used in many Asian countries. Scientists say that it is almost impossible to model due to its complicated physical structure.**

This paper proposes a method to incorporate recorded tabla sound to the MIDI standard so that it can be used like other midi instrument. We have also developed a public web interface to synthesize instrumental music from transcription. We have then added tabla to the music to get a complete performance. The synthesized music is rhythmically perfect and sounds natural having a Mean Opinion Score (MOS) of 4.5 [with a scale from 1 (poor) to 5 (good)] of 4.

Keywords—Music;Tabla; Synthesis; Rhythm; Pitch; Tempo

I. INTRODUCTION

Tabla [Fig. 1] is the prime membranophone percussion instrument of many Asian countries. It is one of the most complex drums in the word involving finely articulated intricate stroke combinations and having 'strong sense of pitch' and intensely developed rhythmic systems. It is really amazing that how such a tiny set of drums produce too many beautiful distinct sounds.

The Nobel Laureate Raman [Raman, 1934] made the first scientific study of this family of drums and concluded that it is almost impossible to model *tabla* due to its complicated physical structure. In spite of that few attempts [Sathej & Adhikari] were made to model this instrument but due to eccentric structure of *bayan* and lack of circular symmetry, they failed to produce realistic sound.

(i) (ii)

Fig. 1. Percussion musical instrument (i) 'Tabla'. The bigger one is 'dayan' and smaller one is 'bayan'. (ii) 'Mridanga' (also called 'Khol')

However, technologies such as MIDI and software *SoundFont* make *sample-based synthesis* possible. We know that MIDI files, instead of storing sounds directly, contain instructions to render them. A SoundFont, on the other hand contains base samples corresponding to certain key of a musical keyboard and other music synthesis parameters such as loops, vibrato effect, and velocity-sensitive volume changing etc. Synthesizers read instruction from MIDI files and render them using sounds stored in the SoundFont.

Unfortunately, Tabla is not in the MIDI instrument list. In this paper, we have shown how we can add tabla to a SounfFont so that it can be used by the MIDI synthesizers like an ordinary instrument. Moreover, non-standard and alternative musical expressions can be achieved while maintaining the performance expression of the traditional Tabla interaction. Since, this instrument has high sense of pitch; it is added as general instruments. We identified a total of 28 (16 for *dayan* and 12 for *bayan*) syllables each spanned over 16 pitch levels. Since MIDI keyboard has 128 keys, $\lceil (28x16)/128 \rceil = 4$ instruments are added to a custom Sound Bank number 100.

We have also developed a system to synthesize instrumental music from given transcription. We have then added tabla to the music to get a complete performance. The synthesized music has a Mean Opinion Score(MOS) of 4.5 [with a scale from 1 (poor) to 5 (good)] of 4.

We have also developed a free-online [t2s-uroy] **I**nstrumental **M**usic **S**ynthesis (IMS) system that can generate Rabindranath Tagore's (A Nobel laureate in literature) Bengali song in specified instruments. Visitors can download synthesized song by selecting transcription and parameters such as rhythms (*taals*), pitch and tempo (*lay*).

II. STATE OF THE ART

Although instrumental music synthesis is easier than synthesizing singing voice, a very few attempts were made.

In [ukr1, ukr2], we already proposed schemes for synthesizing table performance by concatenating signals of basic tabla syllables In this paper, we have shown how we can incorporate it to the MIDI standard so that it can be used like other MIDI instrument and avoid complex signal processing.

The perfect model of *tabla* is not yet known. Raman [Raman, 1934] made the first scientific study of this family of drums. He and his co-workers obtained through a series of experiments, the Eigen modes and Eigen values of the *mridangam.* In [Ramakrishna & Sondhi], authors subsequently modelled the drum but agree with Raman's experimental values to within 10%. The approximate solutions were provided by [B_S_Ramakrishna, Sarojini & Rahman], but the agreement with experimental values is also very poor.

In [Lehana_ Dubey], a method for the separation of *tabla* sound from a mixer of vocal and *tabla* is presented. Although, the separated *tabla* sound didn't contain any residual of vocal sound, the quality of the sound was poor. In [Kapur_Ajay], authors describe the design of a simple electronic Tabla controller (*ETabla*). This is a too simple design and cannot even produce moderately realistic sound; hence cannot be used as professional controller.

In [Li_Jinlong], a morphing algorithm based on Gaussian Mixer Model (GMM) was presented. A singer's timbre can also be added suitably by changing a factor k.

In [Saino], authors proposed a corpus-based singing voice synthesis system based on Hidden Markov Models (HMMs). Although, proposed system can synthesize smooth voice, it still sounds machine like voice.

Electronic *tabla* (Sound Lab's Sangat, Radel's Taalmala Digi-108 and Digi-60Dx, Pakrashi's Riyaz,) can only produce a few pre-synthesized rhythms. Some allow composing new but limited (only 2 to 8) rhythms. These support limited pitch/tempo change and lack adding new rhythm.

III. PROPOSED WORK

The proposed system first generates instrumental song as MIDI file from a transcription file with chosen MIDI instrument (including percussion instrument). Other parameters such as scale, tempo may be customized. Additionally, we can use non-MIDI percussion instrument such as *Tabla*, *Mridanga* [Fig 1.] etc. We have created a SoundFont containing all MIDI instruments as well as these percussion instruments. A synthesizer then takes MIDI file and SoundFont and generates output music as WAV/mp3 file

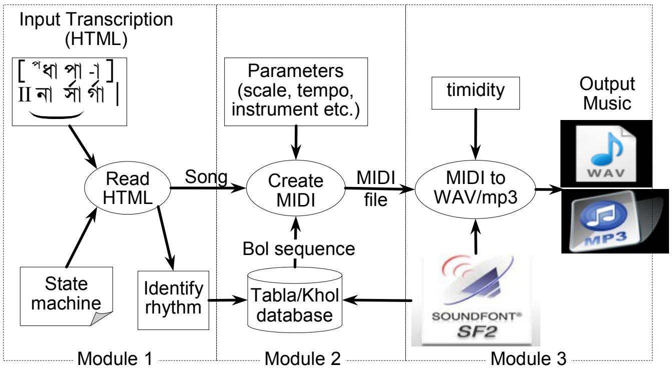

Fig. 2. **I**nstrumental **M**ysic **S**ynthesis (IMS) Framework

The framework of the proposed **I**nstrumental **M**usic **S**ynthesis (IMS) system is shown in Fig. 2. The input is music transcription and output is synthesized instrumental song. The system has basically three modules 1) Read HTML transcription 2) Create MIDI and 3) convert MIDI to WAV/mp3.

IV. READ HTML TRANSCRIPTION

This module reads an HTML transcription file according to a complicated state machine as shown in Fig. 5. To understand the procedure, we must understand the format of the HTML file.

A. *Music Score Representation*

Since different languages use different syntax to write sheet music, uniform representation of musical score (transcription) to be processed by computers efficiently is a great challenge. Although MusicXML [music_XML] is claimed to be a standard open format for exchanging digital sheet music, it is primarily suitable for western music. Many notations in non-western music cannot be represented [Walter_B] by MusicXML. This is very much true for Indian music especially for songs composed by Rabindranath Tagore in *Bengali* language. Tagore, in his more or less 2600 songs, introduced some unique notations, which cannot be represented in MusicXML.

Fortunately transcriptions of Tagore's songs have been digitized [rabindra-rachanabali_nltr] as simple HTML files and are available freely. The transcription is stored as a table as shown in Fig. 3:

নাম	:		আধেক ঘুমে নয়ন চুমে														
							‖										
II	না	র্সা	র্গা	|	র্রা	র্সা	-া	I	না	র্সা	র্রা	|	র্সা	না	-া	I
	আ	ধে	ক		ঘু	মে	◦		ন	য়	ন		চু	মে	◦	
I	ধা	না	না	|	না	না	-া	I	ধা	-পা	-া	|	-মা	-া	-া	I
	স্ব	প	ন		দি	য়ে	◦		যা	◦	◦		◦	◦	য়্	

Fig. 3. An example HTML transcription file of Tagore's song

Note names are stored in transcription file as ordinary English characters. The representation of Bengali notes in English and corresponding MIDI key is shown in Fig. 4

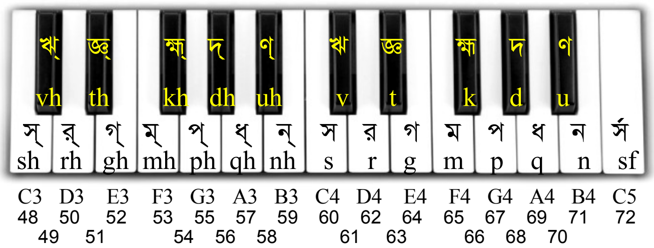

Fig. 4. Indian note to Western note and MIDI key number mapping

B. *State machine*

A summary of Tagore's notation is given below:

{}—Everything between this '{' and '}' is repeated twice.

()—Everything between ‘(‘ and ‘)’ is ignored during repetition. Here is an example.

{ সা –া সদা –া | $^{\text{দ}}$পা –া $^{\text{দ}}$া পা I $^{\text{প}}$মা –া পা –া | (-পণা –ণদা পা সা) }

II—Start and end delimiter of stanza. IIII—End of song. These, at the end of any stanza transfer the control to the beginning of the first stanza (called ‘Asthayi’). Characters ‘I’ and ‘|’ are used to represent a full rhythm and its sections respectively.

[ ]—Alternative transcription written above be the primary transcription. Here is an example: [$^{\text{প}}$ধা পা -রা]
রা সা -া

I[ ]I or II[ ]II—Go to the first stanza and use alternative transcription. Here is an example:

...ধা পা -া **I[ ]I**

The character ‘‖’ on the top of a note tells a musician to either stop completely here or to start a new stanza from here.

‖
II না র্সা র্গা | র্রা র্সা -া I

‘Meend’ (represented by ‘‿’) refers to a glide from one note to another and is used often in vocal and instrumental music. It is written as: জ্ঞা -দা

The state transition diagram for these rules is shown in Fig. 5

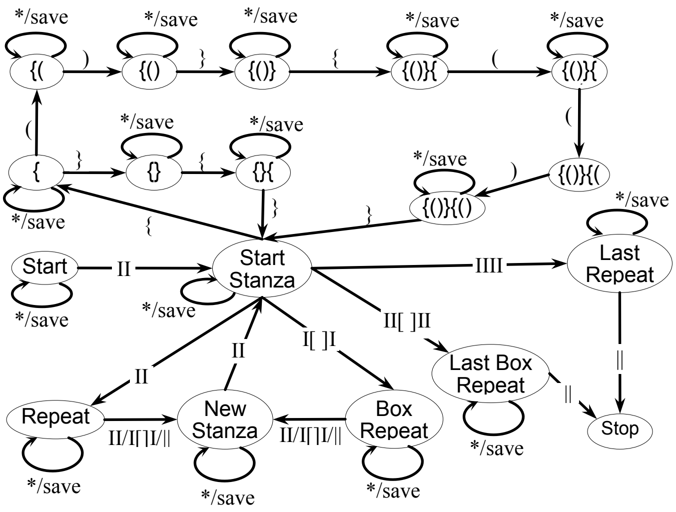

Fig. 5. State machine to read HTML transcription

Given an HTML transcription file and the above state machine, module 1 generates a raw sequence of notes to be processed and played.

V. Create MIDI notes

To generate MIDI notes from raw note sequence, MIDI key, its start time, duration, channel, track, bank etc. are needed. Note name to MIDI key number mapping in C scale is shown in TABLE I. Here note name ‘r’ corresponds to middle C.

The full and half note duration is represented by ‘া’ and ‘ঃ’ respectively. Any other duration is just a combination of them. For example, ‘াঃ’ represents 1.5 notes.

TABLE I. Note name to MIDI key mapping

Note	vh	rh	th	gh	mh	kh	ph	dh	qh	uh	nh	s	v	**r**	t	g	m	k
Key	47	48	49	50	51	52	53	54	55	56	57	58	59	**60**	61	62	63	64
Note	p	d	q	u	n	sf	vf	rf	tf	gf	mf	kf	pf	df	qf	uf	nf	
Key	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	

A unique feature of Tagore’s song is the presence of so called *touch note*, which is played/sung in very small duration (not standardized) compared to a full note. It is written like a superscript before or after prime note(s). Here are some examples:

$^{\text{প}}$মা $^{\text{দ}}$পা$^{\text{ণ}}$ $^{\text{দণ}}$পা ম$^{\text{রজ্ঞ}}$া

We considered the duration of touch note as $1/10^{th}$ of a full note. Here are some examples of timing (shown in []) of the touch note and prime note:

$^{\text{প}}$মা = প[.1] ম[.9]
$^{\text{দ}}$পা$^{\text{ণ}}$ = দ[.1] প[.8] ণ[.1]
$^{\text{দণ}}$পা = দ[.1] ণ[.1] প[.8]
ম$^{\text{রজ্ঞ}}$া = ম[.8] র[.1] জ্ঞ[.1]
গম$^{\text{রজ্ঞ}}$া = গ[.4] ম[.4] র[.1] জ্ঞ[.1]

We shall now take following segment of transcription to demonstrate how MIDI notes are generated from it.

I রসা –া রা -া | রা –পা $^{\text{ম}}$জ্ঞা -া I (1)

It is represented in English (omitting delimiter) as

rsa –a ra -a ra –pa Mta -a (2)

The uppercase letters are used for touch notes. If we calculate the duration of all notes, it looks as:

r[0.5] s[0.5] - r - r p m[0.1] t[0.9] - (3)

The ‘-’ represents the continuation of previous note. So, the second note ‘s’ should continue one full note more; i.e the duration of ‘s’ will be 1.5. The MIDI note matrix for (3) is shown below:

start time	duration	channel	key	velocity	track
0.0	0.5	1	60	127	1
0.5	1.5	1	58	127	1
2.0	2.0	1	60	127	1
4.0	1.0	1	60	127	1
5.0	1.0	1	65	127	1
6.0	0.1	1	63	127	1
6.1	1.9	1	61	127	1

We created a uni-track MIDI files. The channel, track and velocity are selected as 1, 1 and 127 (maximum) respectively.

A. Preparing Tabla/Khol database

We now create MIDI notes for *Tabla*. Traditional Indian music is primarily practice-oriented and the rules of compositions themselves are taught from teacher to disciple, in person. Accordingly, although oral notation for *tabla* stroke names is very developed, written notation is not standardized. Playing table involves extensive use of the fingers and palms in various configurations to create a myriad of different sounds and rhythms, reflected in the mnemonic syllables (*bol*). Fig. 6 shows the names of some basic mnemonic syllables with the striking locations to play them.

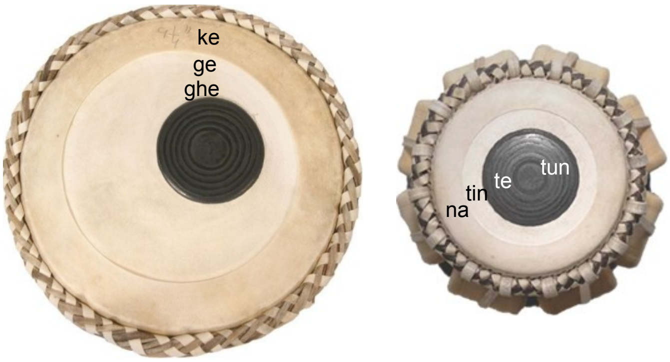

Fig. 6. Positions for plyaing some basic tabla mnemonics

In fact, the set of all syllables is fairly large and requires expertise to understand them. For quick understanding, we have used only seven basic syllables; four for *dayan* 'na', 'te', 'tin' 'tun' and three for *bayan* 'ge', 'ghe', 'ke'.

1) Dayan strokes*:*

ge: holding wrist down and arching the fingers over the *syahi*, the middle and ring-fingers then strike the *maidan* (resonant)

ghe: similar to 'ge' except the heel of the hand is used to apply pressure or in a sliding motion on the larger drum to change pitch during the sound's decay (resonant)

ke or kath: striking with the flat palm and fingers (non-resonant)

2) Bayan stokes*:*

na: striking sharply with the index finger at the rim(resonant)

tin: striking gently with the index finger between *syahi* and rim(non-resonant)

te: striking the center of the *syahi* with the index finger (non-resonant)

tun: striking the center of the *syahi* with the index finger (resonant)

3) Combined strokes*:*

Some syllables are produced by striking both tabla and dagga simultaneously and are often called *combined syllables*. For example, 'dha' is a combination of 'na' and 'ge' where as 'dhin' is a combination of 'tin'and 'ghe'.

Although an experienced table player plays different syllables with different loudness and pitch (for bayan syllables) to increase sweetness, there is no written notation for it. We have developed a notation to represent the syllable, its loudness and pitch in the following form:

syllable.loudness_pitch

For example, ge.2_-3=ge with loudness factor 2 and pitch down by 3. The sign +/- indicates pitch up/down respectively.

TABLE II. SOME POPULAR TABLA TAALS

Name	Beats	Division	Vivhaga
Dadra	6	3+3	X 0
Keherwa	8	4+4	X 0
Tintal (or Trital or Teental)	16	4+4+4+4	X 2 0 3
Jhoomra	14	3+4+3+4	X 2 0 3
Tilwaada	16	4+4+4+4	x 2 0 3
Dhamar	14	5+2+3+4	X 2 0 3
Ektal and Chautal	12	2+2+2+2+2+2	X 0 2 0 3 4
Jhaptal	10	2+3+2+3	X 2 0 3
Rupak (Mughlai/Roopak)	7	3+2+2	X 2 3

1) Rhythms(Taals)

A *tabla* rhythm consists of sequence of syllables (called *bols*) of different durations (1, ½, ¼ note etc.). However, the rhythmic structure (called *taal*) can be quite complex. The basic rhythmic structures can have a large variety of beats (for example 6, 7, 8, 10, 12, 16,..) which are grouped in measures (called *Vivhaga*). TABLE II. shows some of the popular *taals.*

To add tabla to a song we must know the rhythm of the song. This can be identified from HTML transcription file very easily. Note that 'I' and '|' are used to represent a full rhythm and its sections respectively. So, the (1) has a '44' rhythm (called 'Keherwa').

For better accompaniment, a *tabla* player usually plays a variation (called *kaida*) of the rhythm for different sections of a song. So, for each possible rhythm, we maintain a list of bol sequence for each section as follows:

```
22|sthayi=|ghee "na na" |"- ke" "tun na",
      |- "na na" |"- ke" "dha na"|,
      |- "na na" |"- ke" "dha na"|
#
22|repeat=|ghee "na na" |"- ke" "na na"|
#
22|last=|dha.2_-3 "- ghee_3" na "te re",
     | na "- ghee.2_5" na tun,
     |dha.2_-3 "- ghee_3" na "te re"
#
22|antara=$22|sthayi
```

Sections are separated by '#' and takes the following form:

Rhythm|section = comma separated bol sequence

Rhythm and section are separated by '|'. For example, for section 'sthayi' of a song having rhythm '22', |ghee "na na" |"- ke" "tun na" and |- "na na" |"- ke" "dha na"| and |- "na na" |"- ke" "dha na"| are played repeatedly. Since different variations are played, it sounds completely natural.

2) *Processing Rhythms*

To understand how to generate Tabla MIDI notes to accompany a song, let us consider a very simple notation:

|"ghe - te -" "na ke" "- ke" dhin|"tun - - te" "na ke" "- ke" dha|

This is the notation for popular 8-beat (*matra*) rhythms called *keherwa*. Note that this not the basic notation for *keherwa*; but one of its many variations. Since, it uses many syllables, it is good for demonstration. The 8 beats (written as 4/4) consist of two segments (called *bibhag*) each consisting of 4 beats and is delimited by '|' character. A beat may consist of one or more syllables. If a beat has multiple syllables, they are written within " and " characters. A silence is represented by '-' character.

Since, notation uses some characters (such as '|') other than the syllables to divide the entire rhythm into parts, we extract only syllables before processing them further. Following shows the same using only syllables.

'ghe - te -' 'na ke' '- ke' 'dhin' 'tun - - te' 'na ke' '- ke' 'dha'

The timing diagram for the rhythm is shown here.

|ghe| - |te| - | na | ke | - | ke | dhin |tun| - | - |te| na | ke | - | ke | dha |

Time-------------------------------→

This shows that the rhythm consists of 8 beats. The first beat consists of a ghe, a silent, a te and again a silent.

3) *Separation of strikes*

Note that strokes on the *dagga* and *tabla* can be combined, like in the bols : *dha* (*na* + *ge*), *dhin* (*tin* + *ghe*). Due to these characteristics, even if two drums are played simultaneously, the transcription is monophonic - a single symbol is used even if the corresponding stroke is compound. So, it is necessary to separate *tabla* and *dagga* transcription as follows:

```
Initialize t and d as empty sequences;
for each bol in tranacription
      if it is a tabla bol then  append it to t;
      if it is a dagga bol then append it to d;
      if it is a combined bol then
         begin
            [tb, db]=split( bol);
            append tb to t;
            append db to d;
         end
endfor
```

Here is the separated transcription.

Tabla→ '- - te -' 'na -' '- -' 'tin' 'tun - - te' 'na -' '- -' 'na'
Dagga→'ghe - - -' '- ke' '- ke' 'ghe' '- - - -' '- ke' '- ke' 'ge'

4) *Serialization*

We now calculate the duration of each syllables assuming duration of a beat is unity. For example, since the first beat has four syllables, each of the syllables has duration ¼. Following shows the durations of all *tabla* syllables:

-	-	te	-	na	-	-	-	tin	tun	-	te	-	na	-	-	-	na
1/4	1/4	1/4	1/4	1/2	1/2	1/2	1/2	1	1/4	1/4	1/4	1/4	1/2	1/2	1/2	1/2	1

Similarly, following shows the durations of all *dagga* syllables:

ghe	-	-	-	-	ke	-	ke	ghe	-	-	-	-	-	ke	-	ke	ge
1/4	1/4	1/4	1/4	1/2	1/2	1/2	1/2	1	1/4	1/4	1/4	1/4	1/2	1/2	1/2	1/2	1

5) *Normalization*

A notation may contain (probably many) silences. During that time, no striking happens but echo of the previous syllable continues. So, if syllable has one or more silences, the duration for the syllable has to be re-calculated. For example, the following sequence of syllables

ge	-	-	-	-
1/4	1/4	1/4	1/4	1/2

The, duration of each syllable is shown under it. The entire sequence can be replaced by a 'ge' with duration 1.5 (¼+¼+¼+¼+½) units. The number of samples required for this duration is 1.5*FS, where FS is the sampling frequency. If the signal 'ge' has at least those many samples, we take 1.5*FS samples from the beginning. Otherwise, we append necessary zeros at the end of 'ge' signal to make total number of samples equal to 1.5*FS. We call this process as *normalization*. Following shows the normalized duration:

<table>
<tr><td rowspan="2">tabla</td><td>-</td><td>te</td><td colspan="4">na</td><td colspan="2">tin</td><td>tun</td><td>te</td><td colspan="4">na</td><td colspan="2">na</td></tr>
<tr><td>1/2</td><td>1/2</td><td colspan="4">2</td><td colspan="2">1</td><td>1/2</td><td>1/2</td><td colspan="4">2</td><td colspan="2">1</td></tr>
<tr><td rowspan="2">dagga</td><td colspan="3">ghe</td><td colspan="2">ke</td><td>ke</td><td colspan="5">ghe</td><td colspan="2">ke</td><td>ke</td><td colspan="2">ge</td></tr>
<tr><td colspan="3">3/2</td><td colspan="2">1</td><td>1/2</td><td colspan="5">5/2</td><td colspan="2">1</td><td>1/2</td><td colspan="2">1</td></tr>
</table>

Time-------------------------------→

VI. MIDI TO WAV/MP3 CONVERSION

Finally, we convert MIDI file to WAV/mp3 using TiMidity++ synthesizer that requires a SoundFont (.SF2 fie) containing base samples for MIDI instruments and Tabla.

A. *PreparingSoundBank*

A unique feature of the *bayan* is that players can change its pitch by applying pressure on the membrane or sliding their palm across the membrane. This effectively reduces the resonant area which is inversely proportional to pitch and results a strong sense of pitch over a wide range of frequencies. Moreover, the pitch of the *dayan* is tuned according to the scale of the singer/player. So, the syllables for tabla must be added to the SoundFont such a way as if a player is playing the tabla.

TABLE III. SOUND BANK 100

Key range	0-15	16-31	32-47	48-63	64-79	80-95	96-111	112-127
Root key	7	23	39	55	71	87	103	119
Inst 0	de	din	na	ne	open-na	re	sharp-na	sharp-tin
Inst 1	tun	softer-sur	strong-te	sur	te	thap	ti	tin
Inst 2	closed-ge	finger-ke	co-ge	co-ge1	oc-ge	ke	loud-ge	loud-ke
Inst 3	open-ge	open-ke	ge	ghe	-	-	-	-

We recorded 28 syllables (16 for dayan and 12 for bayan) by manually playing tabla at various configuration. Each syllable is spanned over 16 pitch levels so that we have enough space to scale it up/down. Since one MIDI instrument has 128 pitch levels, it can accommodate 8 syllables. Therefore, a total of 4 instruments are required. We choose instruments 0, 1 (for dayan syllables) and 2, 3 (for bayan syllbles) in a separate bank 100. The root key was chosen as the middle of the key range. The key range and root key for syllables is shown in TABLE III. Polyphone SoundFont editor was used to create the SoundFont.

B. *Synthesizing MIDI*

A MIDI synthesizer takes a MIDI file containing a sequence of instruction and a SoundFont and generates a WAV file. We used timidity software synthesizer giving the SoundFont created earlier to it. Since, WAV file occupies significant space; it may be converted to mp3 file.

VII. EXPERIMENTAL RESULT

We could not find any appropriate tabla syllable database. Therefore, we recorded 28 syllables by manually playing tabla at various configuration. In TABLE III. Syllables for Instrument 0 and 1 are for *dayan* and instrument 1 and 2 are for *bayan*. To improve the quality of the sounds, we recorded them in a sound proof lab.

We then created a web-based [http://t2s-uroy.rhcloud.com] interface [Fig. 7] to evaluate the system. The entire system was developed in Java/JSP. Users can choose any of the 2600 songs, select parameters such as pitch, tempo, primary and secondary instruments, table/khol etc. It also provides an transcription editor to enter new transcriptions. The system can generate MIDI/WAV/mp3 file for the song. MIDI file requires the SoundFont having Tabla. The WAV/mp3 files may be listened directly. The generated sounds smooth, rhythmically perfect and natural-sounding.

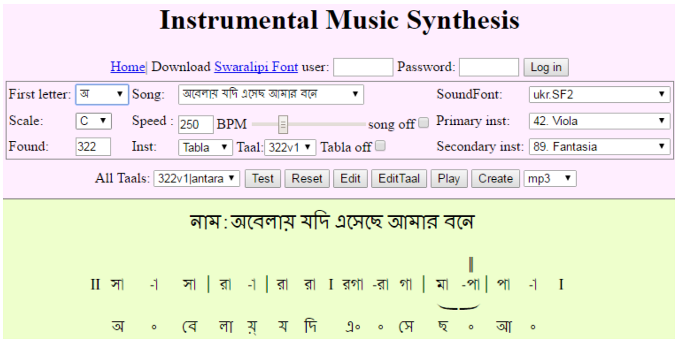

Fig. 7. Interface for Instrumental Music Synthesis

To evaluate the performance of the system, we conducted a subjective listening test. Fifty subjects were asked to rate the naturalness of synthesized instrumental song. They were asked listening at least 10 songs with different parameter settings and with a good quality speaker (especially having bass to get tabla sound properly). The Mean Opinion Score (MOS) was 4.5 on a scale from 1 (poor) to 5 (good). Interestingly, many listeners said that the synthetic music is perfect.

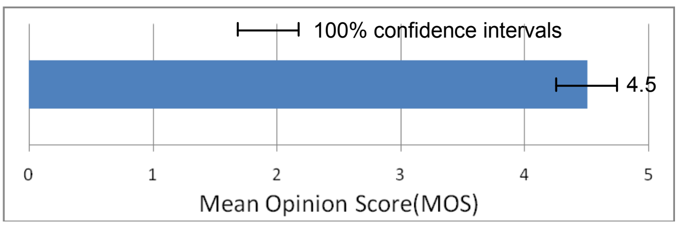

Fig. 8. Evaluation of naturalness

VIII. CONCLUSION

This paper proposed a method to incorporate recorded tabla sound to the MIDI standard so that it can be used like other midi instrument. We have also developed a system to synthesize instrumental music from given transcription. We have then added tabla to the music to get a complete performance. The synthesized music has a Mean Opinion Score(MOS) 4.5 [with a scale from 1 (poor) to 5 (good)] of 4.

References

[1] "T2S: Transcription to Song sysnthesis system", http://t2s-uroy.rhcloud.com[t2s-uroy]

[2] MusicXML Definition, http://musicxml.com/.[music_XML]

[3] Walter B. Hewlett and Eleanor Selfridge-Field, eds., "The Virtual Score: Representation, Retrieval, Restoration", MIT Press, Cambridge, MA, 2001, pp. 113-124. [Walter_B]

[4] Digitized works of Rabindranath Tagore, http://www.rabindra-rachanabali.nltr.org/[rabindra-rachanabali_nltr]

[5] Li, Jinlong, et al. "A Lyrics to Singing Voice Synthesis System with Variable Timbre." Applied Informatics and Communication. Springer Berlin Heidelberg, 2011. 186-193.[Li_Jinlong]

[6] Saino, Keijiro, Heiga Zen, Yoshihiko Nankaku, Akinobu Lee, and Keiichi Tokuda. "An HMM-based singing voice synthesis system." In INTERSPEECH. 2006.[Saino]

[7] Raman, C. (1934). The indian musical drum. In Science, P. I. A., editor, Reprinted in Musical Acoustics: selected reprints, ed. T.D. Rossing Am. Assn. Phys. Tech., College Park, MD, 1988. [Raman, 1934]

[8] B. S. Ramakrishna and M. M. Sondhi, "Vibrations of Indian Musical Drums Regarded as Composite Membranes", The Journal of the Acoustical Society of America 26/4, 523-529 (1954).[Ramakrishna & Sondhi]

[9] T. Sarojini and A. Rahman, "Variational Method for the Vibrations of the Indian Drums", J.Acoust.Soc.Am 30/3, 191-1996(1958).[Sarojini & Rahman]

[10] B. S. Ramakrishna, "Modes of Vibration of the Indian Drum Dugga or Left-Hand Thabala", The Journal of the Acoustical Society of America 29/2, 234-238 (1957).[B_S_Ramakrishna]

[11] Parveen Lehana, Neeraj Dubey & Maitreyee Dutta, "Separation of Tabla from Singing Voice using Percussive Feature Detection", International Journal of Computer Science & Communication, 1/1, 219-222, 2010[Lehana_ Dubey]

[12] Kapur, Ajay, et al. "The electronic tabla controller." Journal of New Music Research 32.4 (2003): 351-359.[Kapur_Ajay]

[13] U. K. Roy, "Concatenated Tabla Sound Synthesis to Help Musicians", Research in Computing Science, 2016 [ukr1]

U. K. Roy, "Composing Recorded Tabla Sound to Accompany Musicians", Proc. Of 2015 Annual IEEE India Conference (INDICON) - Communication & Signal Processing, pp 1-6, DOI: 10.1109/INDICON.2015.7443114 [ukr2]

[14] Fletcher, N.H., & Rossing, T.D. (1998). *The physics of musical instruments*. Second Edition. New York: Springer Verlag.[Rossing][[Fletcher and Rossing, 1998]]

[15] Sathej, G., and R. Adhikari. "The eigenspectra of Indian musical drums." The Journal of the Acoustical Society of America 125.2 (2009): 831-838. [Sathej & Adhikari]

[16] RafAel Ferer, "Timbral Environments: An Ecological Approach to the Cognition of Timbre", Empirical Musicology Review Vol. 6, No. 2, 2011

[17] Mihir Sarkar, Barry Vercoe, "Recognition and Prediction in a Network Music Performance System for Indian Percussion", Proceedings of the 2007 Conference on New Interfaces for Musical Expression (NIME07), New York, NY, USA