A SUPERVISED APPROACH TO HIERARCHICAL METRICAL CYCLE TRACKING FROM AUDIO MUSIC RECORDINGS

Ajay Srinivasamurthy, Xavier Serra

Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
ajays.murthy@upf.edu, xavier.serra@upf.edu

ABSTRACT

A supervised approach to metrical cycle tracking from audio is presented, with a main focus on tracking the tāḷa, the hierarchical cyclic metrical structure in Carnatic music. Given the tāḷa of a piece, we aim to estimate the ākṣara (lowest metrical pulse), the ākṣara period, and the sama (first pulse of the tāḷa cycle). Starting with percussion enhanced audio, we estimate the ākṣara pulse period from a tempogram computed using an onset detection function. A novelty function is computed using a self-similarity matrix constructed using frame level audio features. These are then used to estimate possible ākṣara and sama candidates, followed by a candidate selection based on periodicity constraints, which leads to the final estimates. The algorithm is tested on an annotated collection of 176 pieces spanning four different tāḷas. Though applied to Carnatic music, the framework presented is general and can be extended to other music cultures with cyclical metrical structures.

Index Terms — Rhythm, Musical meter, Carnatic Music, Metrical Cycles

1. INTRODUCTION

Rhythm in many music cultures is organized based on hierarchical metrical cycles at various time scales. The organization and relationship between these cyclical structures forms an important aspect of meter. Tracking these cycles from audio music pieces is an important and useful task for automatic rhythm description and can provide listeners with an understanding of the temporal structure of the music piece at various levels. These rhythmic structures form the basis for building more complex musical patterns and play an important role in defining rhythmic similarity. They are also useful for a rhythm based segmentation of a piece and can be further used for higher level structural segmentation. Most of these metrical cycles can be described using musically well defined periodic rhythmic events at different metrical levels. In this paper, we propose a framework and describe a specific algorithm to track such events, with a primary focus on Carnatic music.

Carnatic music is an art music tradition from South India. It has a long history, a large repertoire, and significant musicological literature. It exists and continues to evolve in the present day social context with a large number of active composers, musicians and listeners. Carnatic music has a well defined rhythmic framework with minimal ambiguity in terminology and practice. It is thus an excellent music culture to explore automatic metrical analysis and tracking, and hence is the primary focus of this paper. We first provide a brief introduction to rhythm in Carnatic music.

This work is partly supported by the European Research Council under the European Union’s Seventh Framework Program, as part of the CompMusic project (ERC grant agreement 267583).

1.1. Rhythm in Carnatic Music

Rhythm in Carnatic music is based on the framework of tāḷa, which provides a broad structure for repetition of music phrases, motifs and improvisations. A tāḷa consists of fixed length time cycles called āvartana which can be referred to as the tāḷa cycle. An āvartana is divided into equidistant basic time units called ākṣaras, and the first ākṣara of each āvartana is called the sama. Fig. 1 shows the structure of a popular Carnatic tāḷa called miśra chāpu in which each āvartana consists of 14 ākṣaras. The ākṣaras are also grouped to give the “beats” of the tāḷa that are displayed by the musician through hand gestures. Sambamoorthy [1] provides a comprehensive description of rhythm in Carnatic music.

The sama of an āvartana is often accented, with notable melodic and percussive events. Most phrase changes and improvisations are aligned with the sama. Hence, local changes in melodic, rhythmic and timbral characteristics are possible indicators of sama. However, many phrases span over many āvartanas and hence it is not necessary that every sama has notable events. Since there is no specific indicator for sams, estimating sama locations is a very challenging task. The primary percussion accompaniment in Carnatic music is the Mridangam, a two-sided pitched drum. Mridangam follows the tāḷa closely and hence Mridangam stroke onsets can be used as the primary indicators of the ākṣaras. Often, the left(bass)-strokes of Mridangam are used to accentuate important events in the progression of the tāḷa.

1.2. State of the art

Automatic rhythm annotation aims to annotate a piece of music with different events and aspects related to its pulsating rhythm. Most of the prior work has mainly focused on onset detection, tempo and beat tracking, downbeat detection, meter estimation, and drum transcription. Müller et al. [2] report the estimation of beat pulsation (beat tracking) and higher-level rhythmic structures such as the measure

Fig. 1: An āvartana of miśra chāpu tāḷa and its structure shown with the terminology used in this paper. The ākṣaras are shown with time ticks, the beats are numbered, and the sama are marked using ×.
length. Approaches such as the one presented by Klapuri et al. [3] aim at estimating structures at several metrical levels at the same time in order to track the metrical grid of a piece. Other approaches focus on specific aspects, such as beat tracking [4], estimating the type of meter [5] in terms of a time signature, or tracking the downbeat locations in a music piece [6, 7], commonly referred to as downbeat detection. Most presented approaches assume the presence of clear cues for a regular metrical grid in the surface rhythm. Hence, they are reported to work reasonably well, especially for European and American popular music. When applied to music with a less clear surface structure, the performance is observed to deteriorate [3, 8]. A recent study that applied the state of the art to Indian and Turkish music showed that analysis methods that exploit the rhythmic structural information are necessary for rhythm description in these music cultures [9].

The work on estimating cyclical metrical structures has been addressed to a very limited extent in Indian Art music. In the context of Hindustani music (Art music culture from North India), approaches for tempo and meter estimation were presented by Gulati et al. [10], transcription of percussive instruments was approached by Chordia [11], and recognition of tāḷas was attempted by Miron [12]. For Carnatic music, Srinivasamurthy et al. [13] proposed to describe meter in terms of the time-span relations at measure, beat and aṣkara levels. None of these approaches have considered the task of tracking the different components of a tāḷa in a piece. To the best of our knowledge, this is the first work in tracking components of a tāḷa.

### 2. Problem Definition

Starting with audio music pieces, we wish to estimate the rhythmic events that will finally lead to a complete description of the hierarchical metrical cycles in the piece. We present a general framework for this task that can incorporate domain specific knowledge about the music and the kind of rhythmic structures present in that music.

We primarily intend to work with large music collections comprising of multiple sources of information. We wish to design an approach that combines all the data sources effectively in the task of rhythm description. Most music collections in Carnatic music contain the editorial metadata of the tāḷa for each piece and hence tāḷa recognition for such a collection is a redundant task. Hence, we propose a supervised approach that uses the tāḷa label (and hence its structure) and then estimates the different events related to the tāḷa in the audio. In this paper, we limit ourselves to estimating the aṣkoras and the samsas, which can be used towards complete tāḷa tracking.

A major component of describing rhythm is the tempo of the piece, measured with the number of aṣkoras per minute (APM). From an audio piece, we estimate an equivalent tempo measure - the aṣkara pulse period (APP) defined as 60/APM. Carnatic music pieces do not have a pre-notated tempo and the musician is free to choose any tempo. Given the APP, since an āvartana has a fixed number of aṣkoras depending on the tāḷa, we can estimate the inter-sama-interval (ISI), the time interval between two adjacent sama pulses or the length of an āvartana. Further, since the pieces are not played to a click track, the APP can vary through a piece and hence we estimate a possibly time varying APP curve from which we can derive a time varying ISI curve.

In summary, we estimate the APP curve, aṣkara and sama time locations from an audio music recording. Though the main task of tāḷa tracking encompasses these sub-tasks, each of these sub-tasks is important in itself and provide useful rhythm descriptors.

### 3. Dataset

The dataset used in this paper is a part of the CompMusic [14] Carnatic music audio collection, which is a research corpus of curated commercial audio releases and metadata with over 397 hours of music and is representative of the present day Carnatic music. Since no rhythm annotated Carnatic audio dataset existed, we compiled a dataset that consists of 176 excerpts of Carnatic music sampled from the CompMusic collection. The pieces span the four most popular tāḷas in Carnatic music in which a majority of pieces are composed. They consist of both vocal and instrumental recordings. The audio is stereo and sampled at 44100 Hz. All the pieces contain a percussion accompaniment, predominantly the Mridangam.

The annotations consist of the editorial metadata about the tāḷa and the time aligned tāḷa annotations including the sama and the beats for each piece. The tāḷa annotations were manually done by a Carnatic percussionist by tapping to the piece and then correcting the taps using Sonic Visualizer [15]. The dataset is representative of the present day performance practice in Carnatic music and spans a wide variety of artists, forms and instruments. The dataset is described in Table 1. From Table 1, we see that though there is no notated tempo in Carnatic music, musicians tend to perform in a narrow range of tempo, as shown by the small variance of ISI. To the best of our knowledge, this is the first tāḷa annotated collection of Carnatic music (The annotations are publicly available for download at [http://compmusic.upf.edu/carnatic-rhythm-dataset](http://compmusic.upf.edu/carnatic-rhythm-dataset)).

### 4. Approach

The primary philosophy of the approach presented in the paper is to incorporate specific knowledge of the rhythmic structures we aim to estimate. For estimating the APP, aṣkara period curve, and the samsas, we first estimate a descriptor computed from the audio that indicates the possible candidates for each musical concept. We then make use of the periodicity and the relationships between these structures to estimate the components. This framework can be generalized to estimating other rhythmic structures by suitably modifying the audio descriptor for the specific music culture and the rhythmic structure under consideration. The algorithm for Carnatic music is

<table>
<thead>
<tr>
<th>Tāḷa</th>
<th># Aṣkara</th>
<th>ISI ± σ_{ISI}</th>
<th>APP ± σ_{APP}</th>
<th># Pieces</th>
<th>LEN</th>
<th>Total Minutes (Hours)</th>
<th># Samas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adi</td>
<td>32</td>
<td>5.38 ± 0.6997</td>
<td>0.1680</td>
<td>50</td>
<td>4.85</td>
<td>252.78 (4.21)</td>
<td>2882</td>
</tr>
<tr>
<td>Rūpaka</td>
<td>12</td>
<td>2.15 ± 0.2508</td>
<td>0.1790</td>
<td>50</td>
<td>4.62</td>
<td>267.45 (4.46)</td>
<td>7582</td>
</tr>
<tr>
<td>Miśra čāpu</td>
<td>14</td>
<td>2.64 ± 0.3394</td>
<td>0.1889</td>
<td>48</td>
<td>6.59</td>
<td>342.13 (5.70)</td>
<td>7795</td>
</tr>
<tr>
<td>Khaṇḍa čāpu</td>
<td>10</td>
<td>1.84 ± 0.2788</td>
<td>0.1840</td>
<td>28</td>
<td>4.41</td>
<td>134.62 (2.24)</td>
<td>4387</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td>176</td>
<td>5.06</td>
<td>996.98 (16.62)</td>
<td>22646</td>
</tr>
</tbody>
</table>

Table 1: The CompMusic Carnatic Music Rhythm Dataset. The table also shows the number of aṣkoras in each āvartana. ISI and APP refer to the median of ISI and APP in the dataset respectively. σ_{ISI} is the standard deviation of ISI in the dataset. The number of pieces, median length of each piece LEN, total size of the dataset and the number of annotated samsas is also shown.
explained in detail in this section.

4.1. Pre-processing

The akṣara pulse most often coincides with the onsets of Mridangam strokes. To enhance the Mridangam onsets, percussion enhancement is performed on the downmixed mono audio, as it has been shown to improve beat tracking performance in pieces with predominant vocals [16]. The predominant melody is estimated using the algorithm proposed by Salamon et al. [17] using which, we extract the harmonic component of the signal using a sinusoidal+residual model [18]. The percussion enhanced signal $x_p[n]$, with the harmonic component suppressed, is used for further processing (Fig. 2(c)).

4.2. Akṣara period and pulse tracking

The spectrogram of $x_p[n]$ is used to compute four frame level spectral-flux based onset detection functions [19] computed every 11.6 ms. The first function ($d_0[m]$) uses the whole frequency range of the spectrogram and the other functions computes the spectral flux only in the range of 0-120 Hz ($d_i[m]$) and captures the low frequency onsets of the left(bass)-side of the Mridangam.

The function $d_a[m]$ is used to compute a Fourier-based Tem- pogram [20] computed every 0.25 second using a 8 second long window (Fig. 2(d)). If we denote the time indices at which the tempogram matrix $M_{i,m}$ is obtained from $x_p[i]$ at index $i$, we can track the most predominant APP curve by estimating the best path $P = \{p_i : i = 1, 2, \cdots, T\}$ through the tempogram matrix $M_{i,m}$, and the local continuity of APP. We define a cost function that is an extended version of the one used by Wu et al. [21] as shown in Eqn. 1.

$$ J_1(P, \theta_1, \theta_2) = \sum_{i=1}^{T} M_{p_i,i} - \sum_{i=1}^{T-1} \left( \theta_1 |p_i - p_{i+1}| + \theta_2 O(p_i, p_{i+1}) \right) $$

The function $O(p_i, p_{i+1})$ is an extra penalty term to penalize tempo doubling and halving between adjacent frames, and the weights $\theta_1$ = 0.01 and $\theta_2$ = 0.001 provide different weights to the three terms.

Based on our observation from the dataset (Sec. 3), the search for the best path through the tempogram is restricted between the range of 120 to 600 APM. The above cost function is solved using a dynamic programming (DP) based approach to obtain an APP curve that is then corrected for tempo doubling/halving errors, if any, to obtain the final curve $P^*$. (Fig. 2(d), $P^*$ is shown as a thick red line).

The akṣara pulse locations predominantly lie on strong Mridan- gam onsets. We estimate the akṣara pulse candidates as the peaks of the function $d_a[m]$. Using these $K$ candidate peaks $\{a_k\}, k = 1, 2, 3, \cdots, K$, with locations $t_k$ and peak amplitude $w_k$, we setup a cost function shown in Eqn. 2 to select the best candidates that provide a balance between the amplitude of these candidates and a periodicity provided the estimated APP. The best set of candidates $\{a'_k\} \subset \{a_k\}$ are estimated using a DP approach (Fig. 2(e)).

$$ J_2(\{a_k\}, \delta) = \sum_{k \in \{1, 2, \cdots, K\}} (w_k + \delta C(t_k, t_{k+1}, P)) $$

The function $C(t_k, t_{k+1}, P)$ is a function that returns an exponentially decaying weight based on the time difference between $t_k$ and $t_{k+1}$ in relation to the local APP, $P_{t_k}$. The parameter $\delta$ = 3 provides a trade-off between the two terms. Using the APP and the tālā information, we can obtain the time varying ISI curve for the piece by multiplying the APP by the number of akṣaras in a cycle of the tālā.

4.3. Sama Tracking

As described earlier, the samas are often associated with significant melodic, rhythmic and timbral changes. In this article, we explore the use of MFCC (mel frequency cepstral coefficients) as features for timbral characteristics. As a detection function for sama ($d_s[m]$), we use a novelty function computed through the diagonal processing of a self similarity matrix [22] constructed using frame level z-score MFCC features from audio (using audio processing library Essentia [23]) as shown in Fig. 2(f). Based on the ISI shown in Table 1, a checkerboard kernel with size of 7, 3, 4, and 3 seconds is used for ādi, rūpaka, misra chāpu and khaṇḍa chāpu respectively so that the novelty function is computed over about an āvartana.

The peaks of the novelty function $d_s[m]$ indicate a significant change of timbre at that time. Starting with the premise that timbral change is an important indicator of sama location, we use the peaks of the novelty function to estimate sama candidates. We explore two methods to estimate the candidates. In Method-A, to uniformly choose sama candidates throughout a piece, we cut the piece...
into segments of length 120, 40, 40 and 30 seconds for ādi, rūpaka, mśra chāpu and khaṇḍa chāpu respectively (~10 āvartanas), and estimate the top five most prominent peaks in each segment of the piece as sama candidates ($\{s_k^t\}$). We also propose another approach, Method-B, for candidate estimation that enforces a periodicity constraint while estimating sama candidates. Starting from the peaks of $d_m$ and estimated ISI curve, for a specific peak, we induce the āvarta cycle starting from it. Based on how many other peaks would support such an induced āvarta is assigned as the weight of the specific peak. We can rank order the peaks using this weight and choose the top ten ranked peaks as the same candidates ($\{s_k^t\}$). We also create two random baseline methods RB-1 and RB-2 to compare the performance. In RB-1, we use a randomly chosen constant ISI between 1-8 seconds, and a random starting time between 0-2 seconds to induce periodic samas. In RB-2, we use the estimated ISI with 10 randomly chosen akṣara locations from $\{a_k\}$ as sama candidates. RB-1 neither uses the ISI, nor the candidate estimation using $d_m$, while RB-2 uses the estimated ISI but not the candidate estimation using $d_m$.

Starting with the sama candidates from either Method-A or Method-B, for each candidate, we induce the āvarta cycles based on local ISI period obtained from the ISI curve. For each seed, we search within the next and previous three estimated cycle periods for onset peaks in $d_m$ that support a sama. If we find a supporting onset, we mark it as a sama and then proceed further with the new estimated onset as the new anchor. We stop the induction from a candidate when it does not lead to such a supporting onset. Hence for each candidate, we obtain an estimated sama sequence. Since all candidates are not necessarily sama locations, though the estimated ISI is right, the sequences can have different offsets. The final step of the algorithm is to shift, align and merge these sequences obtained from each candidate. Starting with the longest sama sequence that has been estimated, we merge the other sequences into this based on maximum correlation between the sequences. The merging of these sequences often leads to many sama estimates concentrated around the true location of sama due to small offsets. Since the left bass onsets on the Mridangam are often strong at the samas, all groups of sama estimates that are closer than 1/3rd of ISI are merged into a single sama estimate aligned with the closest left stroke onset obtained from $d_m$. This forms the final set of sama locations $\{s_k\}$ estimated from the candidates and the onset detection function, as shown in Fig. 2(f) with $\times$.

5. RESULTS AND DISCUSSION

The annotated dataset has annotations only for beats and samas of the piece. From the sama locations, we can obtain the ground truth for ISI curve, and hence the ground truth for APP curve. Since we do not have the ground truth for akṣara locations, we present the results only for APP and sama tracking.

<table>
<thead>
<tr>
<th>Measure</th>
<th>CML (%)</th>
<th>AML (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>APP estimation</td>
<td>81.25</td>
<td>98.86</td>
</tr>
<tr>
<td>APP tracking</td>
<td>80.45</td>
<td>96.26</td>
</tr>
</tbody>
</table>

Table 2: Accuracy of APP and ISI tracking

The performance of APP tracking is measured by comparing the ground truth APP curve with the estimated curve with an error tolerance of 5%. We also report the results for estimation of $\overline{APP}$ computed from the whole APP curve. Further, since there can be tempo doubling and halving errors, we also report the accuracy at the annotated correct metrical level (CML) and then using a weaker measure that allows tempo halving and doubling (AML - allowed metrical levels). The results are presented in Table 2. We see that an acceptable level accuracy is achieved at CML for both $\overline{APP}$ estimation and APP tracking and further, there is not a significant difference between their performances, indicating that the algorithm can track changes in tempo effectively. Even when the APP tracking fails at CML, the algorithm tracks a metrically related APP, as indicated by a high AML accuracy.

<table>
<thead>
<tr>
<th>Variant</th>
<th>P</th>
<th>R</th>
<th>F</th>
<th>I (bits)</th>
<th>Cand. Est.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method-A</td>
<td>29.02</td>
<td>19.01</td>
<td>21.61</td>
<td>1.17</td>
<td>20.46</td>
</tr>
<tr>
<td>Method-B</td>
<td>24.60</td>
<td>20.15</td>
<td>21.51</td>
<td>1.25</td>
<td>27.85</td>
</tr>
<tr>
<td>RB-1</td>
<td>15.51</td>
<td>17.47</td>
<td>13.73</td>
<td>0.40</td>
<td>-</td>
</tr>
<tr>
<td>RB-2</td>
<td>22.76</td>
<td>19.98</td>
<td>20.62</td>
<td>1.11</td>
<td>15.3</td>
</tr>
</tbody>
</table>

Table 3: Accuracy of Sama Tracking. P: Precision, R: Recall, F: f-measure, I: Information Gain. The values are mean performance over the whole dataset expressed in % except for Information Gain, which is in bits. The last column shows the fraction of the estimated sama candidates that are true samas.

For sama tracking, we report the accuracy of estimation with a margin of 7% the annotated TST of the piece. Given the ground truth and the estimated sama time sequence, we use the common evaluation measures used in beat tracking - precision, recall, f-measure and Information Gain [24] to measure the performance. The results are shown in Table 3, which also shows the accuracy of sama candidate estimation. The results for RB-1 and RB-2 show mean performance over 100 and 10 experiments for each piece, respectively.

We see that the performance of sama candidate estimation and sama tracking is poor in general, with samas correctly tracked only in about a fifth of cases. The precision is higher than recall in all cases, and Information Gain is lower than a perceptually acceptable threshold [25]. Both methods perform better than RB-1, but have comparable results with RB-2, with a slightly better f-measure performance (statistically significant in a Mann–Whitney U test at $p < 0.05$). This shows that the estimated ISI is useful for sama estimation, whereas candidate estimation using novelty function is only marginally useful. The poor performance can be mainly attributed to poor sama candidate estimation with either of Method-A or Method-B. This is further substantiated by the fact that Method-B achieves an f-measure of 43.58% and an information gain of 1.70 bits when at least half the estimated candidates are true samas. This clearly shows that the performance of sama tracking crucially depends on sama candidate estimation. There are only four pieces (among all pieces with accurate ISI estimation) in which all the estimated sama candidates are true samas, for which an f-measure of 89.38% and an information gain of 3.51 bits is achieved. This clearly indicates that the novelty function from which the sama candidates were estimated is not a very good indicator of samas, and better descriptors need to be explored.

6. CONCLUSIONS

We presented a general framework for tracking hierarchical metrical cycle from audio and a specific algorithm to estimate the akṣara pulse period, akṣara and sama locations from Carnatic music audio pieces. We also presented a sama annotated Carnatic music dataset that can be used in automatic rhythm annotation tasks. APP tracking performs to an acceptable accuracy for practical applications, while sama tracking is challenging and performs poorly primarily due to poor sama candidate estimation. Though the framework for estimating sama is promising, the novelty function used presently is not a good indicator for sama and hence better audio descriptors need to be explored for the task.
7. REFERENCES


