MIREX 2012 AUDIO BEAT TRACKING SUBMISSION: IBT

João Lobato Oliveira[1,2] **Matthew E. P. Davies**[1] **Fabien Gouyon**[1] **Luis Paulo Reis**[2,3]
[1]Inst. for Syst. and Computer Eng. of Science and Tech. (INESC TEC), Porto, Portugal
[2]Artificial Intelligence and Computer Science Laboratory (LIACC), FEUP, Porto, Portugal
[3]University of Minho, School of Engineering - DSI, Guimarães, Portugal
{jmso,mdavies,fgouyon}@inescporto.pt lpreis@dsi.uminho.pt

ABSTRACT

This extended abstract briefly describes our submission for the "Audio Beat Tracking" task. The proposed system is fully described in [Oliveira et al., "Beat tracking for multiple applications: a multi-agent system architecture with state recovery." IEEE Transactions on Audio Speech and Language Processing, 20(10):1-10, in press, 2012]. The proposed system integrates an automatic monitoring and state recovery mechanism, that applies (re-)inductions of tempo and beats, on a multi-agent-based beat tracking architecture. Beats can be predicted in a causal or in a non-causal usage mode, which makes the system suitable for diverse applications.

1. SYSTEM DESCRIPTION

IBT (standing for INESC Porto Beat Tracker) is the proposed tempo induction and beat tracking algorithm, fully described in [1]. It is inspired by the multi-agent tracking architecture of BeatRoot, where competing agents process parallel hypotheses of tempo and beat [2].

As depicted in Fig. 1, IBT's algorithm follows a top-down architecture composed of: *i)* an audio feature extraction module that parses the audio data into a continuous feature sequence assumed to convey the predominant information relevant to rhythmic analysis; followed by *ii)* an agents induction module, which (re-)generates a set of new hypotheses regarding possible beat periods and phases; followed by *iii)* a beat tracking module, which propagates hypotheses, proceeds to their online creation, killing and ranking, and outputs beats on-the-fly and/or at the end of the analysis. To handle abrupt changes in the musical signal more rapidly and robustly, in real-time contexts (*e.g.*, data streaming), the system also integrates *iv)* an automatic monitoring mechanism (AMM). This mechanism is responsible for supervising the beat tracking analysis of the signal to the necessity of recovering the state of the system through re-inductions of beat and tempo.

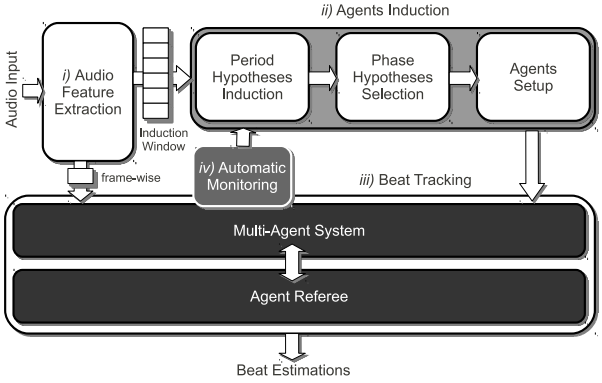

Figure 1. IBT block diagram.

1.1 Audio Feature Extraction

Our implementation makes use of the spectral flux as our mid-level rhythmic representation. The spectral flux measures magnitude variations across all frequency bins, k, of the signal's spectrum, $X(n,k)$, along consecutive analysis frames, n. We compute the time-frequency representation of the signal through a Fast Fourier Transform (FFT), using a Hamming window envelope with w = 1024 samples (23.2 ms at a sampling rate of Fs = 44100 Hz) and 50% overlap. As proposed in [3], the spectral flux is calculated using the L_1-norm over a linear magnitude, which is half-wave rectified, $HWR(x) = \frac{x+|x|}{2}$, to retain only increasing variations in the magnitude spectrum:

$$SF(n) = \sum_{k=-\frac{w}{2}}^{\frac{w}{2}-1} HWR(|X(n,k)| - |X(n-1,k)|). \quad (1)$$

To remove spurious peaks while retaining the most salient, a low-pass second-order Butterworth filter is applied to accumulated SF at every time-step of the analysis. This filter is applied in both forward and reverse directions resulting in an $\tilde{SF} \rightarrow [0, 10]\ s$ window with zero-phase distortion.

1.2 Agents Induction

This module is responsible for (re-)inducing the system's agents with multiple hypotheses of beat positions and tempo. The process makes use of an induction window with fixed-length built of incoming values of spectral flux.

The induction modes of operation are user-definable and

range from `single` to `reset` or `regen`. In `single` mode the induction is only run at the very beginning of the signal's processing to set up the first set of agents. In both `reset` and `regen` modes the system is induced at the beginning of the analysis and whenever requested to be re-induced with new hypotheses of beat and tempo. Moreover, in the `reset` mode all previously existing agents are killed from the system and no continuity is implied on the score of the newly created agents. If the system is operating in `regen` induction mode, all previously exiting agents are kept and the new agents are scored in proportion to the score of the best agent at the time.

1.2.1 Period Hypotheses Estimation

The first step in the agents induction stage is to compute a continuous periodicity function, based on the unbiased autocorrelation function (ACF) of the spectral flux's induction window, $A(\tau)$, along time-lags τ:

$$A(\tau) = \sum_{n=0}^{I} \tilde{SF}(n)\tilde{SF}(n+\tau), \tag{2}$$

where $\tilde{SF}(n)$ is the smoothed spectral flux value at frame n, and I is the length of the induction window. The periodicity function is then parsed by an adaptive peak-picking algorithm to retrieve K global maxima, whose time-lags constitute the initial set of period hypotheses P:

$$\begin{cases} P_i = \arg\max_\tau \left(A(\tau)\right), \;\; i = 1, ..., K \\ A(\tau) > \delta \cdot \frac{rms(A(\tau))}{T} \end{cases}, \tag{3}$$

where δ is a fixed threshold parameter, set to 0.75, and T is the chosen tempo range, at a 6 ms granularity.

1.2.2 Phase Hypotheses Selection

For each of the P_i period hypotheses, M phase hypotheses, ϕ_i^j, (where j is the index of the phase hypotheses for the i-th period hypothesis) are considered among possible phase locations. In order to maximize the suitable starting offsets, these phases are assigned with fixed positions starting at the beginning of the induction window and spaced by $ceil(\frac{P_{max}}{M})$ (where P_{max} is the maximum admitted period) until the end of it. For each period hypothesis, P_i, we generate an isochronous sequence of beats (a "beat train template", Γ_i^j) of constant period for each possible phase, ϕ_i^j, such as $\Gamma_i^j = \phi_i^j + \gamma_i^j P_i \;:\; \gamma_i^j = 0, \cdots, \Upsilon_i^j$, where Υ_i^j is the total numbers of beats in Γ_i^j. Using a simplified tracking procedure, for each P_i we then select the beat train template that best fits the spectral flux represented in the considered induction window. This results in K period-phase, (P_i, ϕ_i), hypotheses and their respective s_i^{raw} scores.

1.2.3 Agents Setup

The final induction step is to compute and rank a score for each hypothesis. At first, and as proposed in [2], in order to favor candidates whose periods present metrical (*i.e.*, integer) relationships with others we defined a relational score, s_i^{rel}, to each agent, given by:

$$s_i^{rel} = 10 \cdot s_i^{raw} + \sum_{\substack{k=0 \\ k \neq i}}^{K} r(n_{ik}) \cdot s_k^{raw}. \tag{4}$$

The s_i^{rel} of each agent considers both the agent's own raw score, s_i^{raw}, weighted by 10, and the raw scores, s_k^{raw}, of all other $K-1$ agents weighted by $r(n_{ik})$:

$$r(n_{ik}) = \begin{cases} 6 - n_{ik}, & \text{if } 1 \leq n_{ik} \leq 4 \\ 1, & \text{if } 5 \leq n_{ik} \leq 8 \\ 0, & \text{if otherwise} \end{cases}, \tag{5}$$

where $n_{ik} = \frac{P_i}{P_k} : P_i >= P_k \vee n_{ik} = \frac{P_k}{P_i} : P_i < P_k$ is the integer ratio between each pair of period hypotheses, (P_i, P_k), with a tolerance of 15%. This weighting factor is intended to favor *single*, *duple*, *triple*, or *quadruple* metrical relationships among the agents' periods, to a maximum of $r(n_{ik}) = 6$. Ultimately, we define the final agents' scores s_i, for the `single` and `reset` induction modes of operation, as:

$$s_i = \frac{s_i^{rel}}{max(s^{rel})} \cdot max(s^{raw}). \tag{6}$$

In the `regen` induction mode of operation these s_i are additionally normalized by the score of the best agent, sb, at the time-frame n_r of the new induction request:

$$s_i = s_i \cdot sb(n_r). \tag{7}$$

The estimated hypotheses, (P_i, ϕ_i, s_i), can now be used to initialize a set of K new beat agents, which will start their beat tracking activity.

1.3 Beat Tracking

1.3.1 Agents Operation

Using the initial (P_i, ϕ_i, s_i) induction hypotheses, an initial set of K beat agents, representing alternative hypotheses regarding beat positions and tempo, will start to causally propagate predictions based on incoming data. Each agent's beat prediction, b_p, is evaluated with respect to its deviation (*i.e.*, $error$) from the local maximum, m, in the observed $\tilde{SF}$ data within a two-level tolerance window around b_p; such that $error = m - b_p$. This two-level tolerance consists of an *inner* tolerance region, $T_{in} \in [b_p - T_{in}^l, b_p + T_{in}^r]$, $T_{in}^l = T_{in}^r = 46.4\,ms$, for handling short period and phase deviations, and an asymmetric *outer* tolerance region, $T_{out} \in [b_p - T_{out}^l, b_p - T_{in}^l[\cup]b_p + T_{in}^r, b_p + T_{out}^r]$, with a left margin $T_{out}^l = 0.2 \cdot P_i$ and a right margin $T_{out}^r = 0.4 \cdot P_i$ (see Fig. 2). Consequently, two alternative scenarios arise. The first scenario corresponds to a local maximum found inside the *inner* tolerance window. In order for the agent's (P_i, ϕ_i) hypothesis to adapt to the observed prediction $error$, the agent's period, P_i, and phase, ϕ_i, are compensated by 25% of that error (limited by the minimum, P_{min}, and maximum, P_{max}, admitted periods):

$$\begin{cases} P_i = P_i + 0.25 \cdot error \\ \phi_i = (\phi_i + 0.25 \cdot error) + P_i \end{cases}, \;\; \exists\, m \in T_{in}. \tag{8}$$

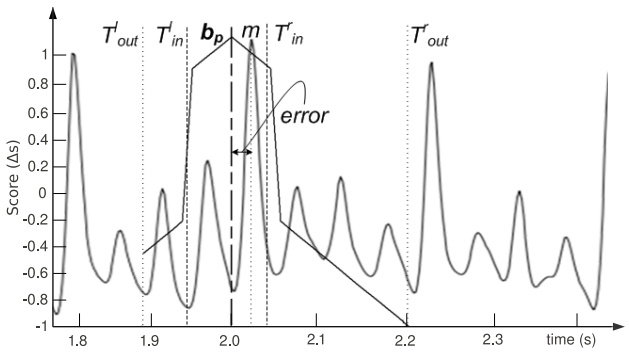

Figure 2. Score function (thin line) around a beat prediction, b_p, with P_i = 120 bpm. Example of local maximum, m, in the spectral flux (thick line) found in the considered inner tolerance window, T_{in}.

The second scenario considers larger deviations, with local maxima in the *outer* tolerance window. In this case, the agent under analysis keeps its period and phase but, in order to cope with sudden variations of tempo and timing, it generates three children $\{C_1, C_2, C_3\}$:

$$C_1 : \begin{cases} P_{C_1} = P_i \\ \phi_{C_1} = (\phi_i + error) + P_{C_1} \end{cases}, \exists\, m \in T_{out}, \quad (9)$$

$$C_2 : \begin{cases} P_{C_2} = P_i + error \\ \phi_{C_2} = (\phi_i + error) + P_{C_2} \end{cases}, \exists\, m \in T_{out}, \quad (10)$$

$$C_3 : \begin{cases} P_{C_3} = P_i + 0.5 \cdot error \\ \phi_{C_3} = (\phi_i + 0.5 \cdot error) + P_{C_3} \end{cases}, \exists\, m \in T_{out}, \quad (11)$$

where $P_{C_1}, P_{C_2}, P_{C_3} \in [P_{min}, P_{max}]$. To remain competitive, these new agents inherit 90% of their parent's current score. Ultimately, different situations may terminate an agent's operation, at any point of the analysis: *replacement*, *redundancy*, *obsolescence*, or *loss*.

1.3.2 Agent Referee

To determine the best agent at each data frame, a central *Agent Referee* keeps a running evaluation of all agents at all times. This is conducted by scoring the beat predictions of each agent with respect to its "goodness-of-fit" for incoming spectral flux data.

The following score function, $\Delta s(error)$, is applied around each beat prediction b_p in order to evaluate the distance, $error$, between b_p and the local maximum, m, in the spectral flux inside either the *inner* or the *outer* window (see Fig. 2):

$$\Delta s(error) = \begin{cases} \left(1 - \frac{|error|}{T^r_{out}}\right) \cdot \frac{P_i}{P_{max}} \cdot \tilde{SF}(m), & \text{if } m \in T_{in} \\ \left(\frac{|error|}{T^r_{out}}\right) \cdot \frac{P_i}{P_{max}} \cdot \tilde{SF}(m), & \text{if } m \in T_{out} \end{cases}. \quad (12)$$

1.3.3 Non-Causal Version

Whereas the causal processing of the system retrieves the beats of the *current* best agent, at any time-frame, in the non-causal version only the *last* best agent is considered. This longterm decision distinguishes the family of agents whose cumulative score prevails for the whole piece. In this way, every agent keeps a history of their beat predictions, attached to the one inherited from its parent, and transmits it to future generations. In the case of a re-induction of the system, all new agents inherit the history of the best agent at the time of the new induction request.

1.4 Automatic Monitoring Mechanism

We created an AMM that looks for abrupt changes in the score evolution of the best agent. This monitoring runs at time increments of $t_{hop} = 1\,s$ and it looks for the variation, $\delta\overline{sb}_n$, of the current mean chunk of measurements of the best score, $\overline{sb}_n$, in comparison to the previous, $\overline{sb}_{n-t_{hop}}$:

$$\delta\overline{sb}_n = \overline{sb}_n - \overline{sb}_{n-t_{hop}} : \overline{sb}_n = \frac{1}{W} \sum_{w=n-W}^{W} sb(n-w), \quad (13)$$

where n is the current time-frame, $W = 3\,s$ is the size of the considered chunk of best score measurements, and $sb(n)$ is the best score measurement at frame n. A new *agents induction* of the system is requested if $\delta\overline{sb}_{n-1} \geq \delta_{th} \wedge \delta\overline{sb}_n < \delta_{th} : \delta_{th} = 0.03$. To ensure the steady state of the analysis the AMM halts for one full induction window before considering new induction requests.

1.5 Practical Use

IBT was developed in C++ in MARSYAS (Music Analysis, Retrieval and Synthesis for Audio Signals), is multiplatform, and is freely available [1] under GPL licensing. We submitted four modes of operation to MIREX2012, executable with the following commands:

```
$ ./ibt input.wav [IBT-C: causal mode (default)]
$ ./ibt -nc input.wav [IBT-NC: non-causal (
    offline) mode]
$ ./ibt -nc -i "auto-regen" input.wav
[IBT-NC-RG: offline mode with AMM and regen induction]
$ ./ibt -nc -i "auto-reset" input.wav
[IBT-NC-RS: offline mode with AMM and reset induction]
```

2. REFERENCES

[1] J. L. Oliveira, M. E. P. Davies, F. Gouyon, L. P. Reis: "Beat Tracking for Multiple Applications: A Multi-Agent System Architecture with State Recovery," *IEEE Trans. on Audio Speech and Language Processing*, Vol. 20, No. 10, pp. 1–10, in press, 2012.

[2] S. Dixon: "Automatic Extraction of Tempo and Beat from Expressive Performances," *Journal of New Music Research*, Vol. 30, pp. 39–58, 2001.

[3] S. Dixon: " Onset Detection Revisited," *In Proceedings of the 9th International Conference on Digital Audio Effects*, pp. 133–137, 2006.

[1] available at `http://smc.inescporto.pt/research/demo_software/`.