A BAYESIAN APPROACH TO DRUM TRACKING

Andrew N. Robertson
Centre for Digital Music
School of Electronic Engineering and Computer Science,
Queen Mary University of London
andrew.robertson@eecs.qmul.ac.uk

ABSTRACT

This paper describes a real-time Bayesian formulation of the problem of drum tracking. We describe how drum events can be interpreted to update distributions for both tempo and phase, and how these distributions can be combined together in a real-time drum tracking system. Our algorithm is intended for the purposes of synchronisation of pre-recorded audio or video with live drums. We evaluate the algorithm of a new set of drum files from real recordings and compare it to other state-of-the-art algorithms. Our proposed method performs very well, often improving on the results of other real-time beat trackers. The algorithm is implemented in C++ and runs in real-time.

1. INTRODUCTION

This paper concerns the problem of real-time drum tracking, which estimates the time-varying tempo and beat locations from drum signals. This is very important for the construction of interactive performance systems in rock and pop music where drum events define the rhythm of the piece. By using the estimated metrical location of beats, a drum tracker can be used to synchronise pre-recorded audio and visual components with musicians or to analyse aspects of the music relative to the beat, as might be required for generative music.

There have been many approaches to the problem of beat tracking on audio files from a database. A comprehensive review of these is given by Gouyon and Dixon [1]. For use in performance systems, we require algorithms that operate in real-time. Autocorrelation-based approaches are used by *DrumTrack* [2] and *Btrack∼* [3], a causal version of Dan Ellis' dynamic programming algorithm [4]. *B-Keeper* [5] is a specialised drum tracker that uses a rule-based approach and adapts system parameters using explicitly coded expert-knowledge. *IBT* [6] is a real-time implementation of the BeatRoot algorithm by Dixon [7] that uses multiple agents, each representing a tempo and phase hypothesis. Seppänen et al. [8] jointly estimate beat and tatum in a system that was of sufficiently low computational cost to be implemented for the S60 smartphone.

Passages featuring complex rhythms such as syncopation, expressive timing or drum fills present a challenge to these algorithms, since these are situations where musical events no longer occur in synchrony with the underlying beat of the music. In traditional approaches to beat tracking, the assumptions often remain hidden, such as the notion that events or high values in the onset detection function correspond to events that are 'on the beat'. Whilst this might often be true, this is not the case with syncopated rhythms, and this results in beat trackers that do relatively well across the board but exhibit difficulties following complex rhythms [9].

Here, we shall adopt a Bayesian formulation approach to the problem. One advantage of this is that the underlying assumptions have to be made explicit. Cemgil et al. [10] have formulated the problem of tempo tracking for music transcription within a Bayesian framework. Bayesian approaches have been used successfully to tackle some related problems. In automatic accompaniment, Grubb and Dannenberg [11] proposed a stochastic method for following a vocalist in which their current location was represented by a probability distribution.

2. ALGORITHM DESCRIPTION

Traditional beat trackers tend to require three stages:

- feature extraction: input to beat tracking systems tends to either be *event-based*, in which the input is a list such as pitch or event onset times, or else they accept '*continuous*' input from an onset detection function which is a frame-by-frame measure of the strength of the extent to which the current audio frame is a note onset. Bello et al. [12] provide a detailed discussion of the various functions that have been employed.
- tempo induction: this process provides an approximation of the tempo and phase, thus initialising the beat tracker.
- beat tracking or following the beat: this is an iterative process by which the algorithm processes features from the audio to update the current beat location and infer the location of the next beat.

Here we will make some simplifying assumptions that allow us to formulate the problem of beat tracking in a Bayesian framework. Firstly, we will be using an event-based approach so that input to the system is in the form of a simple description of a drum event such as 'kick' or 'snare'. Since drum events in rock music are characterised by the stick or beater hitting the skin, this way of representing the signal makes sense intuitively. It might fare less well in, say, jazz music, where drum events can be more diverse in terms of their sonic qualities, but in rock and pop the defined beat means that an event-based representation helps by removing noise from the input and decreasing the amount of computation that is required. We also accept as input the music sequencer's beat events whose tempo is controlled by the output of the algorithm.

We propose two distributions for the random variables representing the beat period and the phase which represent our belief as to the values of these two quantities. These feature in our model as parameters τ, the beat period, and θ the phase of the relation to the sequencer's click, and we wish to update these distributions on the basis of observed drum events. To do so, we shall employ Bayesian reasoning. Supposing we have a model characterised by parameters $\mathbf{w}$, then our uncertainty about the parameters can be expressed as a *prior* probability distribution $p(\mathbf{w})$, that reflects our hypothesis before observing the data D. On observing data D, we can update our uncertainty through the use of Bayes' theorem:

$$P(\mathbf{w}|D) = \frac{P(D|\mathbf{w})P(\mathbf{w})}{P(D)} \tag{1}$$

The term $P(D|\mathbf{w})$ is the conditional probability of observing data D, given the current model parameters $\mathbf{w}$, and is referred to as the *likelihood function*. The term on the left side, $P(\mathbf{w}|D)$ is the *posterior* and is our distribution over the hypotheses given the observation of the data which is what we want to calculate. The term $P(D)$ is the probability of observing the data under all possible hypotheses. This can be considered to be a normalising constant to ensure that the posterior probability will integrate to one, and in practice this often does not need be evaluated.

2.1 Updating the tempo distribution

Music psychologists have identified that humans have preferred tempo rates and broadly there tends to be a preference for an optimal beat interval between 600 and 700 ms [13] with preferred periods ranging between 429 to 725 ms [14]. Here we enforce a strict limitation on the allowed tempi, referred to here as τ_{min} and τ_{max}. We chose a minimum beat period of 360 ms and maximum 800 ms, corresponding to 168 beats per minute (BPM) and 76 BPM respectively. We will describe how to update the beat period hypothesis assuming there is a prior beat period distribution and a current estimate $\hat{\tau}$ between these limits.

Suppose our incoming data, D, is in the form of a drum event at time t_n, specified in milliseconds from the computers' system time. The Bayes formula states that

$$p(\tau|D,\theta) \propto p(D|\tau,\theta)p(\tau) \tag{2}$$

Then for another recent drum event occurring at previous time, t_k, the interval between the two events is $t_n - t_k$ ms. Given our current beat period estimate, $\hat{\tau}$, this interval

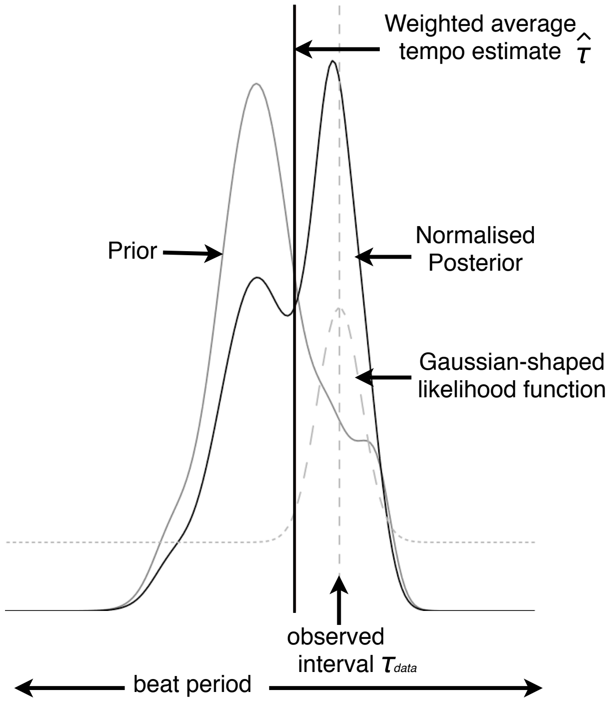

Figure 1. The likelihood function (light grey) and the prior (dotted) give rise to a new posterior tempo distribution (dark grey). The Gaussian shape in the likelihood function is centred on the new tempo observation, τ_{data}, indicated by a light grey vertical line. The new beat period estimate, $\hat{\tau}$, is given by the weighted average over the posterior distribution (vertical dark line).

corresponds to an integer number of beat intervals, given by

$$v(n,k) = \text{round}(\frac{t_n - t_k}{\hat{\tau}}), \tag{3}$$

and thus, where this is positive, our estimate for the beat period from observing drum events at times t_n and t_k is

$$\tau_{data} = \frac{t_n - t_k}{v(n,k)} \tag{4}$$

Our *likelihood function* can be computed directly from this estimate given some model assumptions which we must describe explicitly. We assume that the actual tempo gives rise to noisy observations, so that our likelihood function, the probability distribution of the actual beat period given this sole observation, is a mixture of a Gaussian distribution of standard deviation σ_T centred around the observed estimate τ_{data} and white noise across all possible values of τ. This gives

$$p(\tau|\tau_{data}) = \nu_T + (1-\nu_T)\frac{1}{\sigma_T\sqrt{(2\pi)}}\exp(\frac{-(\tau_{data}-\tau)^2}{2\sigma_T^2}), \tag{5}$$

where ν_T is the quantity of noise across all possible values. We expect that there will be a relatively high amount of drum events, none of which we wish to place too much emphasis upon. In order to do so, we chose the value 0.7 for ν_T, suggesting that 70% of the time our observed data point is unrelated to the actual beat period. We also wish to pick an appropriate balance for σ_T (set by hand to $\frac{\tau_{max}-\tau_{min}}{32}$). If it is to high, the Gaussian will be wide and the tempo will not be picked out sufficiently, whereas if it is too low, the Gaussian will be shaped like a delta peak and there will be less contribution to values for τ close to but different from τ_{data}. Since we want successive observations that are close to a central value to reinforce each other, we require a balance between being too specific and not giving enough weight to the data.

Our posterior distribution is calculated through the application of the Bayes formula, as stated in Equation 2. We take the product of our prior distribution and our likelihood function, and then normalise. Finally, our beat period estimate $\hat{\tau}$ can be calculated as the integral of the posterior over all possible values of τ, so that

$$\hat{\tau} = \int_{\tau_{min}}^{\tau_{max}} \tau p(\tau)\,\mathrm{d}\tau \tag{6}$$

where $p(\tau)$ is the posterior distribution for the beat period. An alternative is to take the maximum aposteriori probability (MAP) estimate. Finally, we also model subtle changes in the underlying tempo that may occur during the performance by adding Gaussian noise around the beat period estimate $\hat{\tau}$. Thus at successive time steps between drum events, we also update our posterior distribution:

$$p(\tau) \leftarrow p(\tau) + n(0, \sigma^2_{noise}) \tag{7}$$

Sequential use of the Bayes' theorem mean that the posterior will be used as our prior when the next data observation is made and the process repeats iteratively.

2.1.1 *Tempo Initialisation*

In order to initialise the system in the absence of a count-in or a reliable prior approximation for the tempo, we use the same technique as above but choose a uniform prior that reflects our lack of knowledge about the tempo. The result on iteratively updating the posterior is that within a couple of bars the distribution tends to peak around the correct tempo, or occasionally a multiple of the tempo such as double or half. In performance, we would most likely use an approximation for the tempo and phase to initialise the algorithm with more suitable prior distributions representing our knowledge about these variables.

2.2 Phase Estimation

Since we intend to use the drum tracker to synchronise accompaniment, it makes sense that there is a unique tempo, of beat period $\hat{\tau}$, that is the outcome of the tempo tracking process described in Section 2.1 above. In estimating the phase of the sequencer, we make the assumption that this tempo estimate is correct and events are only analysed for phase *relative to that tempo*. This way, we can treat phase independently and have only one distribution which can be updated using a similar methodology as employed for tempo. Whilst it would be formally more correct to treat the two variables as forming a joint distribution, in practice the tempo distribution tends towards a sharp peak around the correct underlying tempo and so the assumption is reasonable. If phase were treated differently, then this would raise the problem of keeping track of a phase distribution for each tempo, which would incur significant computational costs. Many beat trackers work by finding the optimum tempo and then outputting the phase estimate at that tempo, although some approaches to beat tracking jointly analyse tempo and phase, for example as found in Eck [15] and Dixon [16]. Thus, we may say that

$$p(\theta|D, \hat{\tau}) \propto p(D|\theta, \hat{\tau})p(\theta, \hat{\tau}) \tag{8}$$

On observing a new drum event at time t_n, we first need to update our posterior distribution to account for the time interval between these events. As this interval increases, we expect uncertainty as to the exact tempo and noise within the drummer's internal timing to result in increasing uncertainty as to the beat location. This can be modelled using the tempo distribution. The interval since the last update is $t_n - t_{n-1}$ ms, where t_{n-1} is the time of the previous drum event measured by the computer's system time. At our tempo estimate $\hat{\tau}$, it corresponds to a specific number of beat intervals given by:

$$v_n = \frac{t_n - t_{n-1}}{\hat{\tau}} \tag{9}$$

Then at a given phase θ, the contribution from phases around θ can be calculated using the tempo distribution to account for how the various possibilities of phases and tempos might combine together. To understand this, consider how an earlier phase at a slower tempo would predict the same beat location as a later phase at a faster tempo. So, for a 1ms difference in beat period, the interval of v_n beat periods creates a time difference of v_n ms in phase offset, which translates as a difference of $\frac{v_n}{\hat{\tau}}$ in terms of relative phase. Since the arrays are recorded discretely, we give the formulation as a sum rather than the integral. Then

$$p(\theta) = \sum_k p(\hat{\tau} + kd_T)p(\theta - \frac{v_n k d_T}{\hat{\tau}}) \tag{10}$$

where d_T is the time interval between adjacent bins in the tempo array (we used 240 bins for the beat period range 360 to 800 ms, so $d_T = 1.83$). Thus, if for notational purposes here, we define ψ_n as $\frac{v_n}{\hat{\tau}}$, then the phase at θ has a contribution of $p(\theta)p(\hat{\tau})$ from the tempo estimate $\hat{\tau}$, a contribution $p(\theta - \psi_n d_T)p(\hat{\tau} + d_T)$ from the tempo of period $\hat{\tau} + d_T$, a contribution $p(\theta + \psi_n d_T)p(\hat{\tau} - d_T)$ from the tempo of period $\hat{\tau} - d_T$, and there is a contribution $p(\theta - 2\psi_n d_T)p(\hat{\tau} + 2d_T)$ from the beat period $\hat{\tau} + 2d_T$, and so on. We then renormalise the array to get our updated prior distribution at time t_n. This now reflects how uncertainty in the tempo distribution translates into uncertainty in phase given the interval between two most recently observed events. We now wish to calculate our likelihood function, $p(\theta|D)$. The drum event happens at time t_n, which corresponds to a phase θ_n. If the closest beat time (recent or predicted) is at $\xi(n)$ ms relative to the computers system time (where the n is included to acknowledge the dependance upon t_n), then the relative phase is given by

$$\theta_n = \frac{t_n - \xi(n)}{\hat{\tau}} \tag{11}$$

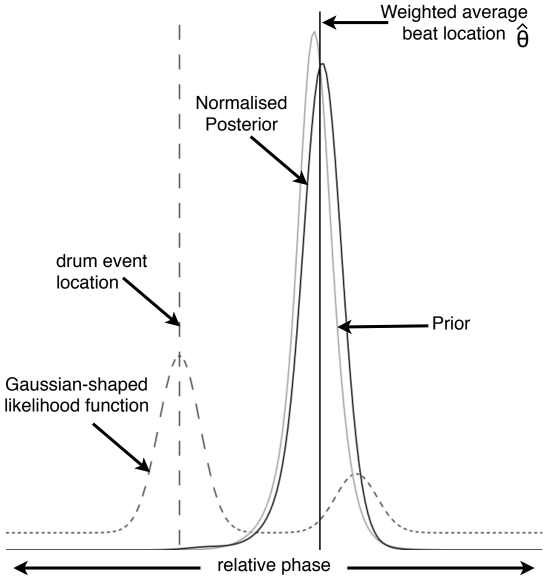

Figure 2. The posterior (dark grey) is calculated as the product of the prior distribution (light grey) and the likelihood function (dotted). In this case, the event is most probably an early sixteenth since it is approximately a quarter of the beat period to the left of our phase estimate, labelled $\hat{\theta}$.

Where the event occurs on the beat, we will model this as a combination of a Gaussian centred on the events location relative to the beat estimate and noise across all possible phases. So, given the observed drum event at relative phase θ_n, the likelihood function is

$$p(\theta|\theta_n) = \nu_P + (1 - \nu_P)g(\theta, \theta_n, \sigma_P) \tag{12}$$

where the Gaussian contribution is

$$g(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{(2\pi)}}\exp(\frac{-(x-\mu)^2}{2\sigma^2}) \tag{13}$$

However, this also depends upon the drum event type, the drum pattern and the location of the event. Only if we interpreted *every* event as being on the beat, would we wish to use the formulation of Equation 12. To understand how this is so, consider an event happening on an eighth note. In this case, we should not necessarily model the likelihood function as a Gaussian around the beat location, since if we believe it to be an eighth note, then it would suggest that the beat is half a period before and after the drum event. Thus, our likelihood function ought to be dependent upon metrical position in the bar.

Here, we will use a methodology that assumes we have an reasonable estimate for the phase and see how this allows us to interpret the rhythmic features. To do so, we divide the region between beats into twelve regions of equal duration with the first region, labelled 0, which corresponds to time $\xi(n)$. We measure the phase relative to the computer's click so that $\xi(n)$ always has a phase of zero. Whilst the predicted beat time $\xi(n)$ is not necessarily at the same relative phase as our optimal estimate $\hat{\theta}$ (since we may be in the process of adjusting), in practice the two will be very close since the system constantly makes adjustments as its output so that, according to the current tempo and phase estimate, $\hat{\theta}$ is predicted to be zero in the near future.

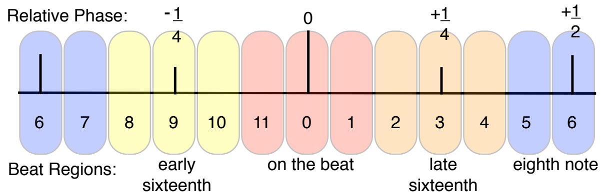

Figure 3. Phase is defined relative to the computer's click track (at time $\xi_(n)$) which has phase 0. Events happening in the different regions are interpreted differently, resulting in different Gaussian mixtures in the likelihood function.

We can say that we would expect events happening in region 0 close to the expected beat location at time $\xi(n)$ and the adjacent regions, labelled 1 and 11, to correspond to events that are 'on the beat'. Thus these events have a likelihood function as given in Equation 12. Events in regions 5, 6 and 7 might constitute events at beat times, but when if our beat estimate is reasonable, it is more likely that they correspond to eighth note events that happen an eighth note after the actual beat. Thus we represent the likelihood for these events as a mixture of two Gaussians, one around where the event happened and one an eighth note away. In this case,

$$p(\theta|\theta_n) = \nu_P + (1-\nu_P)[\alpha g(\theta, \theta_n, \sigma_P) + (1-\alpha)g(\theta, \theta_n + \frac{1}{2}, \sigma_P)] \tag{14}$$

where the Gaussian function, g, is as specified in Equation 13. Through testing we chose a value of alpha of 0.3. In the two region groups 2, 3 and 4, and 8, 9, 10, which we interpret as potentially being sixteenth note events that occur after and before the beat respectively, we use the likelihood function given by

$$p(\theta|\theta_n) = \nu_P + (1-\nu_P)[\alpha g(\theta, \theta_n, \sigma_P) + (1-\alpha)g(\theta, \theta_n \pm \frac{1}{4}, \sigma_P)] \tag{15}$$

where the sign in the second Gaussian is minus for events occurring after the beat and plus for before and alpha is again 0.3. In Figure 2, we show an event that uses this mixture of noise with the remainder one Gaussian weighted by 70% around the event and one Gaussian a quarter of a beat period later weighted by 30%, due to the chance that it is an early sixteenth note. We found in practice that whilst the balance is helpful to stabilise a beat interpretation, weighting the sixteenth too highly would inhibit the tracker to respond to variation in tempo and result in too high a stability at alternative phase estimates. The need for a balance between inertia and reactivity has been commented on by Gouyon and Dixon [1] as a common feature of rhythm description systems. We adjusted the value for likelihood noise ν_P by hand and chose 0.56.

2.2.1 Update Rule

Having updated both distributions, we estimate that the underlying beat period is $\hat{\tau}$ and the relative phase is $\hat{\theta}$. This

Drummer (piece)	Human Tapper	Proposed Bayesian	B-Keeper	Btrack∼	CFM
Adam Whitfield (Dropped D)	26.4	**19.1**	23.6	23.3	-
Whetham Allpress (BlitzKrieg)	18.3	**11.8**	19.5	23.1	21.1
Whetham Allpress (Cannibal Island)	**15.3**	15.4	52.9	20.4	-
Jem Doulton (Funky Riff#07)	30.1	14.7	**14.2**	20.2	19.6
Mark Heaney (Syncopated Beats piece I)	**20.7**	47.8	37.1	42.1	36.5
Mark Heaney (Syncopated Beats piece II)	**17.7**	58.0	35.6	73.9	-
Al Pickard (Funk)	20.2	18.3	**17.6**	22.9	-
Hugo Wilkinson (Follow The Leaders)	**17.5**	18.0	20.7	23.2	23.4
Rod Webb (Reggae beat)	15.3	22.3	**15.1**	26.3	45.6
Adam Betts (Hum1)	**20.1**	25.4	-	50.3	20.9
Adam Betts (Swung loop)	25.9	27.1	-	29.1	**20.2**
David Nock (Speed up drums)	16.4	**12.5**	34.7	20.6	21.9
Marcus Efstratiou (Billy Jean)	**14.5**	15.9	21.7	21.6	16.7
Metronome (120 BPM)	17.3	5.2	**2.1**	-	12.3

Table 1. Average absolute error (ms) between the output of beat tracking algorithms and drum events from the percussive onset detector *bonk*∼ that are 'on the beat'. The time in bold is the closest synchronisation to the ground truth (as defined by the drum beats) for each piece. Those marked '-' failed to track the beats and lost synchronisation.

phase error corresponds to $\hat{\theta}\hat{\tau}$ ms. The single beat period sent as output to our sequencer is $\hat{\tau} + 0.6\zeta$, where ζ is the estimated remaining phase error (reset to the new estimate $\hat{\theta}\hat{\tau}$ after every event). On each beat, a new period is sent out, so we update:

$$\zeta \leftarrow 0.4\zeta \tag{16}$$

Subsequent information in the form of new drum onset events leads the process repeats iteratively.

3. EVALUATION

We have evaluated the algorithm on audio files of drum recordings, made either in band rehearsals or during the development and testing of the drum tracking software system *B-Keeper* [5]. These files have the characteristic of being at relatively steady tempi, so that there are no sudden shifts, although a couple do change tempo slowly over the course of half a minute or so. We have collected the beat locations predicted by four algorithms: the proposed Bayesian tracker, Stark et al.'s *Btrack* [3], our previous algorithm *B-Keeper* [5], and a comb filter matrix beat tracker (CFM) described in Robertson et al. [17], based on methods similar to Eck's Autocorrelation Phase Matrix [15]. We also collected the annotations made by the author using an Oxygen MIDI keyboard and tapping with a cowbell sound.

In some cases, we allowed manual adjustments to ensure that each algorithm was initialised to the correct beat period and phase. *B-Keeper* requires an approximation of the tempo for initialisation. *Btrack*∼ tends to find the right tempo without any intervention, suggesting it is a reliable option when manual intervention is not allowed. The comb filter matrix beat tracker requires two consecutive taps at the beat period to indicate the tempo and phase. In the case of our proposed algorithm, we initialised the algorithm with a uniform prior and waited for it to find the correct tempo. In the few cases where the phase estimate was on the off-beat, we reset the phase distribution to be uniform so that the algorithm would be able to find the correct phase estimate.

In order to evaluate the algorithms, we require ground truth annotations for where the beat locations are. We maintain that drum events define the beat where they occur on the 'one', 'two', 'three' and 'four' of the bar. Whilst we might expect some expressive playing, all trackers are at the same disadvantage in predicting such shifts in phase. We made use of the expert ability of human tappers to understand meter and respond to changes in tempo and phase to allow us to automatically label those drum events which would be the ground truth if one annotated by eye. The audio from the kick and snare drum microphones was passed through the *bonk*∼ object [18], a fast onset detector for percussive sounds. Where these times were within 80 ms to the human annotated beat locations, the onset times were understood as defining the beat locations and thus used as ground truth in the evaluation. This method differs from conventional beat tracking analysis where human annotations are taken as ground truth on the basis that the drums define the beat and so clear drum onsets can be used instead.

We computed the average absolute error between kick and snare events and the beat time where these fall close to the beat. The results are presented in Table 1. This gives an indication of how well each drum tracker has predicted the drummer's actual beat. We have found that our proposed method performs very well with respect to other state-of-the-art algorithms. In many cases, it gives comparable results to *B-Keeper* and on six tracks out of fourteen (often the more conventional rhythmically speaking) it is closer than the human tapper. Whilst *Btrack*∼ behaves reliably, the mean offset is higher than our proposed algorithm for all but one track. This might be due to a latency issue since where the tracking was successful, we observed our method to have a mean error of roughly zero (with no bias before or after the beat), whilst *Btrack*∼ tended to have positive error between 10 and 20 ms. We also found evidence of the negative asynchrony as described in Ascher-sleben [19], whereby human subjects tend to tap *before* the beat and this asynchrony varied between 0 and 20 ms.

4. CONCLUSIONS AND FUTURE WORK

Firstly we remark on two ways in which our system might be improved.

- Rhythm Interpretation: Since we have a relatively stable way in which to track small changes in the beat period and phase, we might explicitly model and learn the rhythmic pattern. This might extend our ability to handle tempo shifts or recover from errors, since if we are able to interpret events correctly, then new events could convey more reliable information than the current formulation permits.

- Learning Probabilities: Whilst we have used parameter settings made by hand, such as the standard deviations of the Gaussians that contribute to the likelihood functions for phase and period, we might be able to estimate these from the data. In addition, we might estimate the ratio between noise and reliable estimates. Approximations to these from analysis of drum data might improve the performance of the system or be the default starting parameters as observations of a particular drummer's style might allow the system to learn more optimal values.

We have presented a real-time Bayesian algorithm for the tracking of tempo and phase in drum signals. This is evaluated on real signals from drum tests and we find that the algorithm compares favourably with the state-of-the-art. With a view to supporting the community's aims for Reproducible Research, the algorithm, the code required to test it and the drum files are available online [1] .

5. REFERENCES

[1] F. Gouyon and S. Dixon, "A review of automatic rhythm description systems," *Computer Music Journal*, vol. 29, no. 1, pp. 34–54, 2005.

[2] N. Collins, "DrumTrack: Beat induction from an acoustic drum kit with synchronised scheduling," in *Proceedings of International Computer Music Conference*, 2005.

[3] A. M. Stark, M. E. P. Davies, and M. D. Plumbley, "Real-time beat-synchronous analysis of musical audio," in *Proc. DAFX-09*, 2009, pp. 299–304.

[4] D. P. W. Ellis, "Beat tracking by dynamic programming," *Journal of New Music Research*, vol. 36, no. 1, pp. 51–60, 2007.

[5] A. Robertson and M. Plumbley, "B-Keeper: A beat-tracker for live performance," in *Proc. International Conference on New Interfaces for Musical Expression (NIME), New York, USA*, 2007, pp. 234 – 237.

[6] J. Oliveira, F. Gouyon, L. G. Martins, and L. P. Reis, "IBT: A real-time tempo and beat tracking system," in *International Conference on Music Information Retrieval*, Utrecht, 2010, pp. 291–296.

[7] S. Dixon, "Evaluation of the audio beat tracking system beatroot," *Journal of New Music Research*, vol. 36, no. 1, pp. 39–50, 2007.

[8] J. Seppännen, A. J. Eronen, and J. Hiipakka, "Joint beat and tatum tracking from music signals," in *International Conference on Music Information Retrieval ISMIR 2006*, 2006, pp. 23–28.

[9] R. B. Dannenberg, "Toward automated holistic beat tracking, music analysis and understanding," in *International Conference on Music Information Retrieval*, 2005, pp. 366–373.

[10] A. T. Cemgil, H. J. Kappen, P. Desain, and H. Honing., "On tempo tracking: Tempogram Representation and Kalman filtering." *Journal of New Music Research*, vol. 28, no. 4, pp. 259–273, 2001.

[11] L. Grubb and R. B. Dannenberg, "A stochastic method of tracking a performer," in *Proc. International Computer Music Conference*, 1997, pp. 301–308.

[12] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. Sandler, "A tutorial on onset detection in music signals," *IEEE Transactions on Speech and Audio In Processing*, vol. 13, no. 5, Part 2, pp. 1035–1047, 2005.

[13] J. London, "Cognitive constraints on metric systems: Some observations and hypotheses," *Music Perception*, vol. 19, pp. 529–550, 2002.

[14] A. Semjen, D. Vorberg, and H.-H. Schulze, "Getting synchronized with the metronome: Comparisons between phase and period correction," *Psychological Research*, vol. 61, no. 44-55, 1998.

[15] D. Eck, "Beat tracking using an autocorrelation phase matrix," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2007, pp. 1313–1316.

[16] S. Dixon, "Automatic extraction of tempo and beat from expressive performances," *Journal of New Music Research*, vol. 30, pp. 39–58, 2001.

[17] A. Robertson, A. M. Stark, and M. D. Plumbley, "Real-time visual beat tracking using a comb filter matrix," in *Proc. International Computer Music Conference*, 2011.

[18] M. Puckette, T. Apel, and D. Zicarelli, "Real-time audio analysis tools for Pd and MSP," in *Proceedings of International Computer Music Conference, San Francisco.*, 1998, pp. 109–112.

[19] G. Ascherschleben, "Temporal control of movements in sensorimotor synchronization." *Brain and Cognition*, vol. 48, no. 1, pp. 66–79, 2002.

[1] http://www.eecs.qmul.ac.uk/andrewr~