

FACULTAD DE INGENIERÍA DE LA  
UNIVERSIDAD DE LA REPÚBLICA

PROCESAMIENTO DIGITAL DE SEÑALES DE  
AUDIO

CURSO 2012

---

Beat Tracking

---

*Autores:*

Gonzalo GUTIÉRREZ

Matías TAILANIÁN



14 de diciembre de 2012

---

# Introducción

---

Al escuchar música una reacción inconsciente muy común es mover el pie golpeando el piso a tiempo con el **beat**. La tarea computacional que intenta replicar ese comportamiento es conocida como **beat tracking**.

La métrica de la señal se puede pensar como una estructura de pulsos percibidos a diferentes escalas temporales en una pieza musical. Se consideran 3 niveles métricos básicos: el *tatum*, el *tactus* o *beat* y el *compás*. El *tatum* es el valor en tiempo más pequeño que puede encontrarse en una pieza musical, es la unidad atómica de la pieza. En general los otros valores de duración presentes en la pieza son múltiplos de *tatum*. El *tactus* o *beat* está más relacionado con el aspecto semántico, y está directamente vinculado con el **tempo** de la pieza. Por último el *compás* está vinculado con la tasa de cambios armónicos o la duración de un patrón rítmico.

El problema del seguimiento de *beat* o *beat tracking* es un problema todavía abierto, donde se sigue intentando con gran actividad lograr mejorar los resultados del estado del arte. Tiene un atractivo muy importante en sí mismo pensando en aplicaciones como el acompañamiento automático, asistencia a la hora del editado, estudios musicológicos, efectos de música adaptativos, o seguimiento del ritmo de una batería tocando en vivo, pero además es una parte imprescindible de algunos sistemas más complejos para aplicaciones como etiquetado de música, reconocimiento automático del género, análisis de similitud de música y para lograr la *transcripción automática de música*.

El presente trabajo presenta un algoritmo de seguimiento de *beat* de una pieza musical y se basa en el trabajo de “João Lobato Oliveira, Fabien Gouyon, Luis Gustavo Martins, Luis Paulo Reis”, titulado “IBT: A real time tempo and beat tracking system, presentado en la 11<sup>a</sup> *International Society for Music Information Retrieval Conference, ISMIR*, en 2010” ([1]). Dicho trabajo se basa a su vez en el sistema **BeatRoot**, presentado en ([2]): Dixon S., “Automatic extraction of tempo and beat from expressive performances.”, *Journal of New Music Research*, 2001. De [2] se toma la idea de varios **agentes** compitiendo y llevando varias hipótesis de tempo y fase paralelamente, se agrega robustez ante entradas ruidosas y se implementa en tiempo real. Aún realizando todas las operaciones de forma causal y en tiempo

real, se logra obtener resultados comparables con el estado del arte y se convierte además en el primer software *open source* de seguimiento de beat en tiempo real.

---

## Metodología

---

Como se puede ver en la figura 1.1 el algoritmo consta de 3 fases fundamentales:

- **Audio Feature Extraction:** En esta etapa se transforma la señal de audio en 1 secuencia continua que caracteriza la información más relevante para el análisis rítmico. En esta etapa se basarán las siguientes partes del algoritmo.
- **Pre-Tracking:** Al finalizar el Pre-Tracking se tendrá un conjunto de hipótesis iniciales con respecto a posibles períodos y fases de los beats. Consta de 3 subetapas donde se estimarán las características que definen a un agente:
  - Período
  - Fase
  - Puntaje
- **Beat-tracking:** En esta etapa se propagan las hipótesis y se crean, matan y puntúan agentes.

A su vez se presenta un sistema de evaluación que decide en función del puntaje de cada agente, cuál es el más adecuado para la pieza musical: el **Agent Referee**.

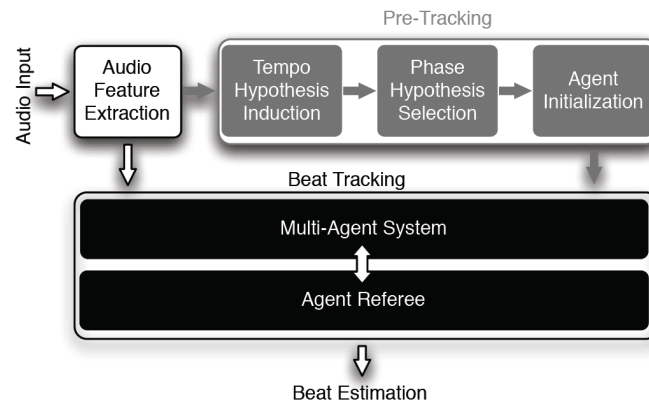


Figura 1.1: Diagrama de bloques

## 1.1. Audio Feature Extraction

En [3] se evalúan distintas magnitudes para caracterizar la información más relevante para el análisis rítmico. Está fuera del alcance del presente trabajo analizar la posibilidad de evaluar distintas magnitudes, por lo que se trabajará con el **Flujo Espectral** como magnitud característica de la pieza musical.

El flujo espectral utiliza una representación tiempo-frecuencia de la señal basada en la transformada corta de Fourier utilizando una ventana de Hamming,  $w(m)$ .

$X(n, k)$  representa al  $k$ -ésimo bin de frecuencia del  $n$ -ésimo frame, como se ve en la ecuación 1.1

$$X(n, k) = \sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1} x(hn + m)w(m)e^{-\frac{2\pi jmk}{N}} \quad (1.1)$$

El flujo espectral lleva una medida del cambio en magnitud de cada *bin* de frecuencia restringido a cambios positivos y sumado a lo largo de todos los *bins* de frecuencia. En la ecuación 1.2 se muestra su ecuación matemática para el cálculo.

$$SF(n) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} H(|X(n, k)| - X(n-1, k)) \quad (1.2)$$

donde  $N = 2048$  (46 *ms* a una frecuencia de muestreo  $r = 44100Hz$ ) y el tamaño del salto es  $h = 441$  (10 *ms* o 78 % de solapado).  $H(x)$  es un rectificador de media onda:  $H(x) = \frac{x+|x|}{2}$ .

## 1.2. Pre-Tracking

La etapa del Pre-Tracking es fundamental ya que en ella se basarán las etapas siguientes del algoritmo. Se analiza una *ventana de inducción* de 5 segundos y se obtiene como salida un conjunto de agentes caracterizados por  $(P_i, \phi_i, S_i)$  para cada agente  $i = 1 \dots N$ . Consta de 3 sub-etapas donde se calcula el período ( $P_i$ ), fase ( $\phi_i$ ) y puntaje ( $S_i$ ) de cada agente  $i$ .

### 1.2.1. Período

La primera etapa consiste en hallar una función de periodicidad continua basada en la autocorrelación y el Flujo Espectral:

$$A(\tau) = \sum_{n=0}^m SF(n)SF(n+\tau) \quad (1.3)$$

La segunda etapa consiste en la detección de eventos, por medio de un hallado y filtrado de picos de la función de autocorrelación descrita anteriormente (ecuación 1.3). Los eventos detectados en  $A(\tau)$  corresponderán con los períodos iniciales de los agentes del Pre-Tracking. El criterio propuesto para obtener dichos picos es:

$$\begin{cases} P_i = \arg \max_i \{A(\tau)\}, & i = 1, \dots, N \\ A(\tau) > \delta \frac{rms(A(\tau))}{M} \end{cases}$$

$\delta$  es un umbral determinado empíricamente en 0,75 y  $M$  es un rango de tiempos definido entre [50,250] BPM.

### 1.2.2. Fase

Para cada  $P_i$  estimado se generan  $j$  hipótesis para la fase:  $\phi_i^j$ . Se supone fase y períodos constantes en cada ventana de análisis.

Se utiliza un *template* de tren de pulsos isócronos para ver cuál ajusta mejor a los picos del Flujo Espectral en la ventana de inducción.

Utilizando un proceso de tracking simplificado como veremos en la sección 1.3 se selecciona el tren de pulsos que mejor ajusta los eventos detectados. Hasta este momento tenemos una pareja  $(P_i, \phi_i)$  para cada agente.

### 1.2.3. Puntaje

A cada pareja  $(P_i, \phi_i)$  se le asigna un puntaje preliminar  $S_i^{raw}$  que corresponde a la suma de los errores entre los elementos del tren de pulsos y los máximos locales del Flujo Espectral, dado por la ecuación 1.5.

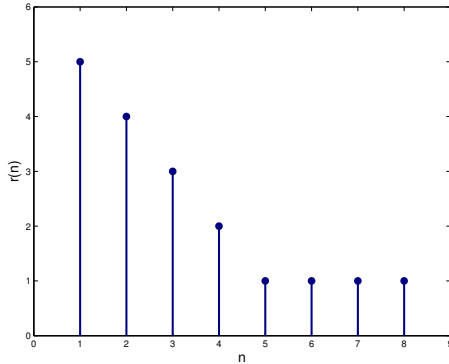


Figura 1.2:  $r(n)$

Luego considerando relaciones métricas entre pares de hipótesis de período  $(n_{ij})$ , se actualiza el puntaje a:

$$S_i^{rel} = 10S_i^{raw} + \sum_{\substack{j=0 \\ j \neq i}}^N r(n_{ij}) S_j^{raw}$$

donde

$$r(n) = \begin{cases} 6 - n, & 1 \leq n \leq 4 \\ 1, & 5 \leq n \leq 8 \\ 0, & \text{en otro caso} \end{cases}$$

que también se puede ver representada en la figura 1.2. El resultado de aplicar la función  $r(n)$  es favorecer todos los agentes que tengan múltiplos enteros uno de otro.

Por último el puntaje final está dado por:

$$S_i = \frac{S_i^{rel}}{\max S_i^{rel}} \max S_i^{raw} \quad (1.4)$$

Tenemos entonces un conjunto de agentes, donde cada uno de ellos está caracterizado por una tríada  $(P_i, \phi_i, S_i)$  que serán utilizados como hipótesis iniciales para la etapa de **Tracking**.

### 1.3. Beat-Tracking

La idea de la etapa de Tracking es supervisar flujo de entrada de los picos detectados del Flujo Espectral y mantener un buen balance entre inercia y rapidez de la respuesta ante cambios.

Cada predicción es evaluada respecto de su desviación al máximo local correspondiente en los datos observados, considerando una ventana como se la mostrada en la figura 1.3, donde se consideran 2 intervalos de tolerancia:

- *inner*:  $T_{in} \in [T_{in}^l, T_{in}^r]$  con  $T_{in}^l = T_{in}^r = 46,4ms$  para manejar pequeñas desviaciones de fase y período
- *outer*:  $T_{out} \in [T_{out}^l, T_{in}^l] \cup [T_{in}^r, T_{out}^r]$  con  $T_{out}^l = 0,2 P_i$  y  $T_{out}^r = 0,4 P_i$  para contemplar eventuales cambios repentinos de tiempo. La asimetría refleja una mayor tendencia a disminuir el tiempo.

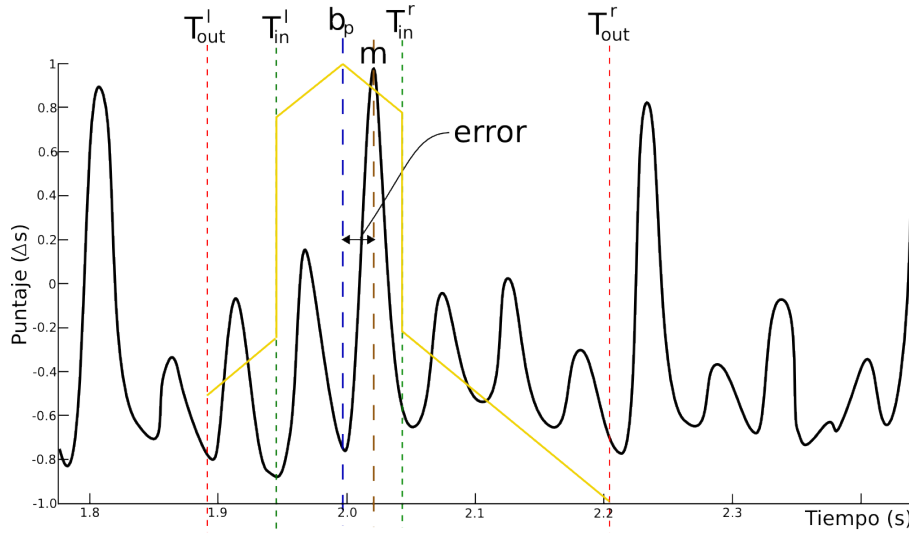


Figura 1.3: Niveles de tolerancia

En consecuencia aparecen 2 claros escenarios distintos: o bien que el máximo local del Flujo Espectral se encuentre en la región *inner*, o bien que se encuentre en la región *outer*.

- Máximo local en región *inner*

En este caso se considera que el agente está siguiendo los beats de la pieza de buena manera y lo único que se realiza es un ajuste fino de período y fase según sigue:

$$\begin{cases} P_i = P_i + 0,25 \text{ error} \\ \phi_i = \phi_i + 0,25 \text{ error} \end{cases}$$

- Máximo local en región *outer*

Este caso corresponde a desviaciones mayores que el caso anterior, donde el agente mantiene su período y fase pero para hacer frente a variaciones de tempo repentinas crea 3 “hijos” ( $C_1, C_2, C_3$ ) para seguir 3 alternativas posibles combinando variaciones de *timing* (fase) y *tempo* (período), según sigue:

$$\begin{aligned} C_1 : & \begin{cases} P_{C_1} = P_i \\ \phi_{C_1} = \phi_i + error + P_{C_1} \end{cases}, \exists m \in T_{out} \\ C_2 : & \begin{cases} P_{C_2} = P_i + error \\ \phi_{C_2} = \phi_i + error + P_{C_2} \end{cases}, \exists m \in T_{out} \\ C_3 : & \begin{cases} P_{C_3} = P_i + 0,5 error \\ \phi_{C_3} = \phi_i + 0,5 error + P_{C_3} \end{cases}, \exists m \in T_{out} \end{aligned}$$

Para que los hijos tengan competitividad respecto al resto de los agentes, se los inicializa con un puntaje igual al 90 % del puntaje del “padre” en ese instante.

### 1.3.1. Matado de agentes

En cualquier punto del análisis, algunas situaciones dan por finalizada la operación de un agente, con los criterios que se explican a continuación:

- *Replacement*: un agente es matado si llegado al límite de agentes (fijado en 30), es el peor de todos y su puntaje es menor al de otro agente creado más recientemente.
- *Redundancy*: para mejorar la eficiencia del algoritmo un agente es matado si está duplicando el trabajo de otro con mayor puntaje. Por duplicación de trabajo se entiende que la diferencia entre los períodos no debe ser mayor a 11,6 ms y la diferencia entre las fases no debe superar los 23,2 ms.
- *Obsolescence*: un agente es matado si la diferencia entre su puntaje con el mejor agente es mayor al 80 % del mejor puntaje
- *Loss*: un agente es matado si parece estar “perdido”, sugerido una cantidad (fijada en 8) de veces consecutivas que la predicción del beat esté por afuera de la región *inner*.

### 1.3.2. Agent Referee

En la versión causal el *referee* va determinando en cada instante cuál es el mejor agente. Mantiene una continua evaluación y le va asignando un puntaje a cada agente respecto a qué tan bueno es el matcheo entre los datos que van llegando y la predicción del beat. La variación de puntaje en cada instante está dada por la ecuación 1.5.

$$\begin{cases} \Delta s = \left(1 - \frac{|error|}{T_{out}^r}\right) \frac{P_i}{P_{max}} SF(m), & \exists m \in T_{in} \\ \Delta s = - \left(\frac{|error|}{T_{out}^r}\right) \frac{P_i}{P_{max}} SF(m), & \exists m \in T_{out} \end{cases} \quad (1.5)$$

$b_p$  es la predicción del beat,  $m$  el máximo correspondiente del Flujo Espectral, y  $P_{max}$  el período máximo permitido (correspondiente a 250 BPM). El cociente  $P_i/P_{max}$  es utilizado para normalizar la función puntaje por el período  $P_i$  y es utilizado como



una manera de disminuir el puntaje de los agentes más rápidos, que de otra manera tendería a aumentar dado que tienen una mayor cantidad de predicciones de beat. La utilización de un puntaje negativo cuando la predicción cae fuera de la región *inner* infiere una penalización al agente correspondiente.

---

## Resultados

---

### 2.1. Implementación

En esta sección se presentan algunos detalles en los cuales fue necesario ahondar el estudio para la correcta implementación del algoritmo presentado en la sección 1, y que resultan interesante destacarlos y documentarlos.

A la hora de realizar el tracking se plantea la posibilidad de pararse en los picos del sF o pararse en las predicciones de beat. centrar la ventana

no matar agentes fuera del outer

El filtrado de los picos se realiza analizando las distancias entre cada máximo y los mínimos aledaños a él.

la matanza de agentes esta pensada para correr en tiempo real. Como nuestro trabajo no lo requería aunque era una buena cosa para conservar, se pueden modificar los criterios de matanza para dejar vivos mas agentes, porque no nos importa que demore un poco mas...

Para la versión no causal se puede utilizar un referee elegido al final del proceso de tracking, como se propone en [2], donde se asume que al ir teniendo buenos puntajes a lo largo del tracking, el agente que tenga mayor puntaje al finalizarlo será el mejor agente global.

### 2.2. Señales sintéticas

Como primer prueba para testear el algoritmo implementado se generan diversas señales sintéticas con período constante. Para ello se convolucionan el sonido de un platillo de batería con un tren de pulsos de frecuencia constante. La prueba siguiente, también sintética consiste en generar una señal con frecuencia variando linealmente a lo largo del tiempo para analizar la performance del algoritmo en el tracking del período variante de una señal.

Como primer ejemplo se presenta el caso de una señal de período constante en 90 *BPM*. Los resultados se muestran en las figuras 2.1.

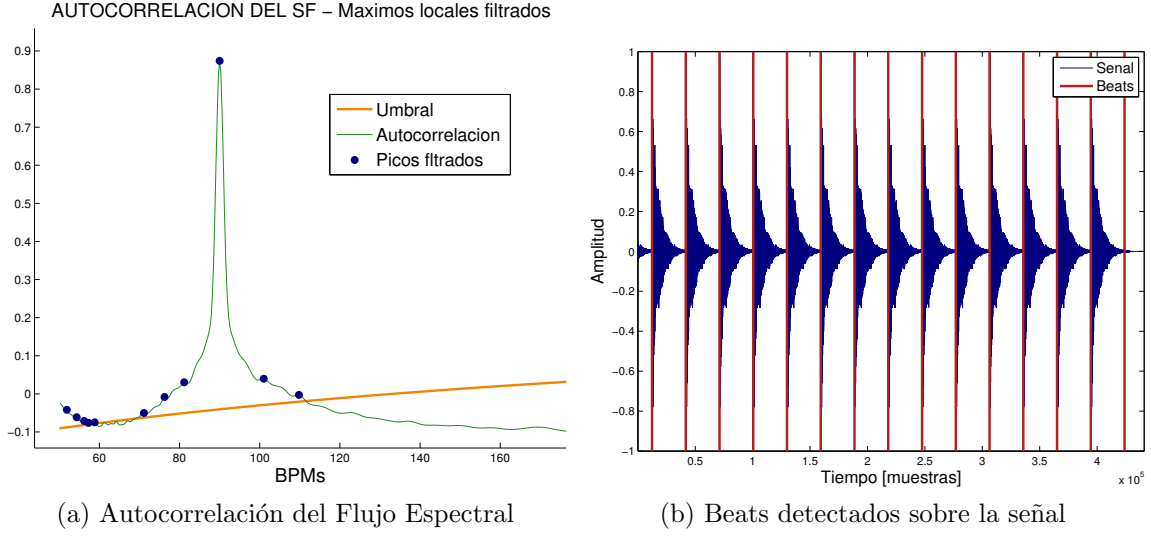


Figura 2.1: Señal sintética de 90 BPM

La figura 2.1a muestra la autocorrelación del flujo espectral calculado en la *ventana de inducción* mencionada en la sección 1.2 del Pre-Tracking. Se puede apreciar claramente que el pico marcado que aparece en la figura corresponde con un período de 90BPM, coincidiendo perfectamente con el período de la señal sintética generada. Se puede decir entonces que el Pre-Tracking captó perfectamente el período del beat. Se muestra también el umbral utilizado para filtrar los máximos locales del Flujo Espectral detectados. Por otro lado en la figura 2.1b se muestra una representación de la señal de audio, y sobrepuesta a ella una línea roja en la posición de cada beat detectado. Se puede corroborar el perfecto funcionamiento del algoritmo en este caso.

Para analizar la capacidad de adaptación del algoritmo a una pieza musical con tempo variante se genera una señal sintética con período variante entre 90BPM y 100BPM. Los resultados se muestran en la figura 2.2.

En la figura 2.2a se muestra la señal sintética original con los beats sobrepuestos, donde se puede observar a grandes rasgos que se realiza el tracking en forma exitosa. Por otro lado en la figura 2.2b se puede observar con más detalle la evolución del tempo a lo largo del tiempo. Comparando ambas curvas se puede afirmar que, a menos de un error de alrededor de 0,5 *BPM*, se realiza un tracking exitoso del tempo de la pieza.

Una vez realizadas las pruebas sintéticas y corroborado el funcionamiento en situaciones controladas, se procede a testear el algoritmo implementado en situaciones reales, como se pasa a describir en la siguiente sección.

## 2.3. Señales reales

Si bien el pasaje de casos sintéticos a reales siempre conlleva el agregado de una cantidad de no idealidades de la señal que pueden causar situaciones no previstas

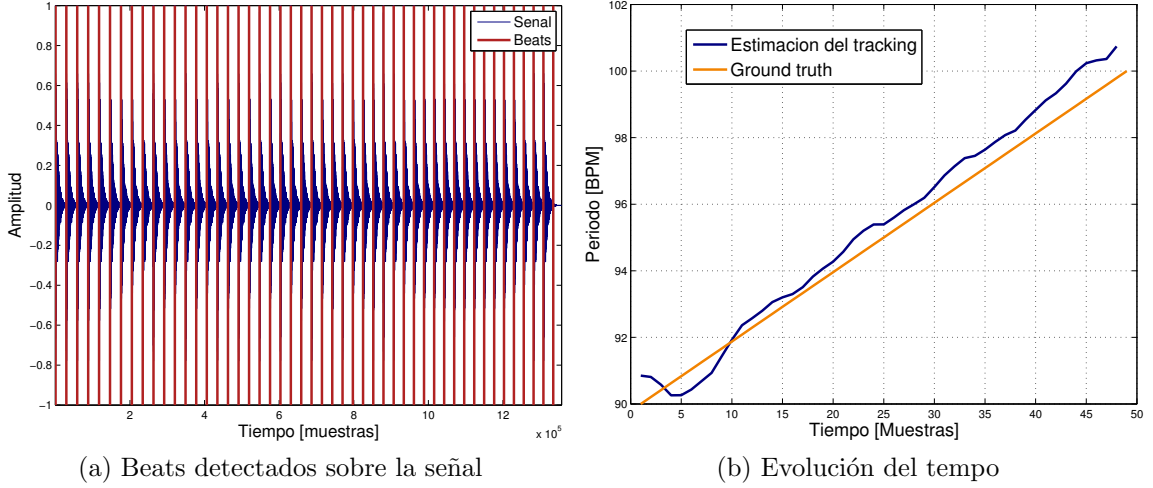


Figura 2.2: Señal sintética con tempo variante en forma lineal

y así un mal desempeño del algoritmo, en este caso la performance alcanzada continúa siendo considerablemente buena. De todas formas se presentan casos donde el algoritmo funciona de forma correcta y donde tiene alguna dificultad.

### 2.3.1. Caso de buen funcionamiento

En esta sección se presentan los resultados para una pieza musical donde el comportamiento del algoritmo coincide completamente con el *ground truth* de la señal.

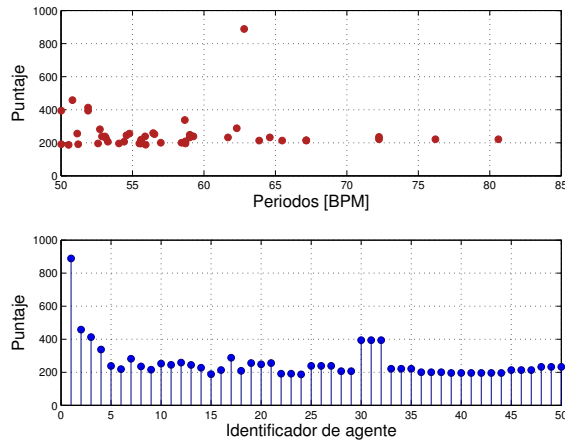
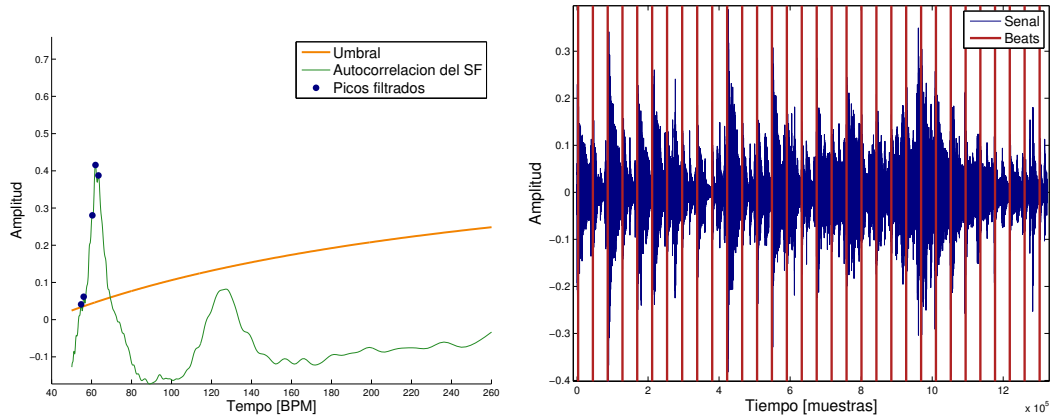


Figura 2.3: Agentes finales

En la figura 2.3 se muestra la configuración de los agentes al terminar el tracking sobre toda la pieza musical. Cada punto en la gráfica representa un agente. En la gráfica de arriba se muestra el puntaje final de cada agente en función de su período final, mientras que en la gráfica de abajo se muestra lo mismo pero en función del identificador de cada agente. El número de agente da una idea de la “edad” de cada agente. Se puede observar en dicha figura que el 1<sup>er</sup> agente tiene un puntaje ligeramente superior que el resto, por lo que será agente ganador para esta pieza.

En la figura 2.4 se muestran más resultados obtenidos para la pieza musical en cuestión. En la figura 2.4a se muestra la autocorrelación del Flujo Espectral en la *ventana de inducción* en función del período en BPM y en la figura 2.4b se muestra la señal original con los beats detectados sobrepuestos.

Si se observa el período del agente ganador mostrado en la figura 2.3 y se lo compara con el *tempo* del mayor pico de la autocorrelación mostrado en la figura 2.4a



(a) Autocorrelación del Flujo Espectral y filtro de picos (b) Beats detectados sobre la señal de audio original

Figura 2.4

se advierte una gran similitud, lo cual sugiere que el agente ganador viene de una familia inicializada por un agente creado en el pre-tracking correspondiente a dicho pico. En este caso el pre-tracking logró estimar de muy buena manera el tempo de la pieza musical y el agente se mantuvo durante toda la pieza ajustando levemente su *tempo* y *timing* y acumulando un buen puntaje.

Por otra parte se puede corroborar visualmente observando la figura 2.4b que en cada ataque de la pieza musical se presenta un beat detectado, lo cual corrobora el buen funcionamiento del algoritmo.

Se realizaron además algunas pruebas cuantitativas utilizando las medidas *Continuity based* y *F-Measure* obteniendo el siguiente resultado:

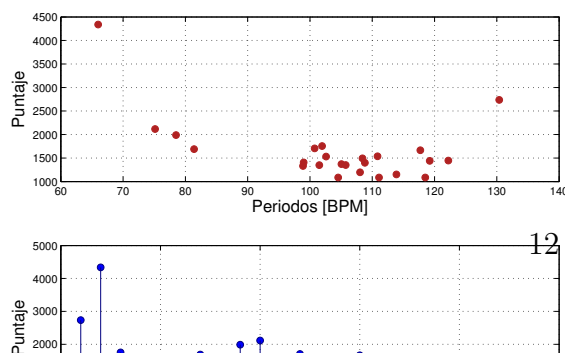
Performance:

Cont-Based:	cC:	100.00	cT:	100.00	aC:	100.00	aT:	100.00
F-Mesure:	f:	94.44	p:	94.44	r:	94.44	a:	89.47

Por último y no menos importante es interesante destacar el resultado auditivo de la señal de salida, el cual parece comportarse de manera perfecta a lo largo de la pieza musical.

### 2.3.2. Caso de funcionamiento comprometido

En esta sección se presenta un caso en el que el desempeño del algoritmo tiene algunas dificultades, aunque en líneas generales la performance resulta aceptable.



En la figura 2.5 se muestra la configuración de los agentes al finalizar el tracking sobre toda la pieza musical. Se advierte fácilmente que hay 2 agentes que tienen

un puntaje ligeramente superior al del resto y que sus tempos son uno aproximadamente el doble del otro: el agente 1 con tempo 130 BPM y el agente 2 con tempo 65 BPM.

He aquí un problema típico mucho más general que su aplicación a este caso de estudio. Es un problema ocasionado al tratar de computar automáticamente en una computadora algunas características que son intrínsecamente semánticas y por tanto humanas. Incluso el seguimiento del beat puede variar dependiendo de la persona, mientras que algunos pueden marcar un determinado tempo, otros pueden marcar el doble o la mitad. En este caso el consenso tomado del *ground truth* dice que el agente que mejor captura el tempo de la pieza musical es el agente número 1. Como salida del algoritmo implementado el mejor agente resulta ser el agente número 2.

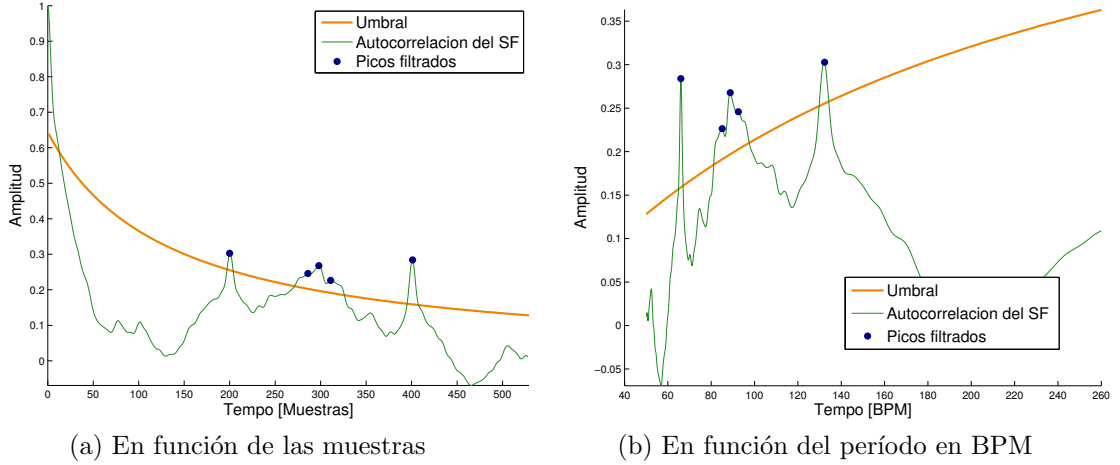


Figura 2.6: Autocorrelación del Flujo Espectral

En la figura 2.6 se muestra la autocorrelación del Flujo Espectral en la ventana de inducción del pre-tracking. Se puede observar que tanto el agente 1 como el 2 son inicializados en esta etapa ya que se advierten 2 claros picos, uno cerca de 65 BPM y otro cerca de 130 BPM.

Por último en la figura 2.7 se muestran los resultados obtenidos si se elige como mejor agente al número 2 (figura 2.7a), y si se elige al número 1 (figura 2.7b).

En este caso es muy difícil darse cuenta visualmente observando las imágenes de la figura 2.7 cuál de los dos sigue de mejor forma al beat, pero si analizamos auditivamente los resultados habrá consenso entre la mayoría de las personas que el agente número 20 sigue de mejor forma el beat de esta pieza musical.

Si se elige como mejor agente al número 2 la performance alcanzada será:

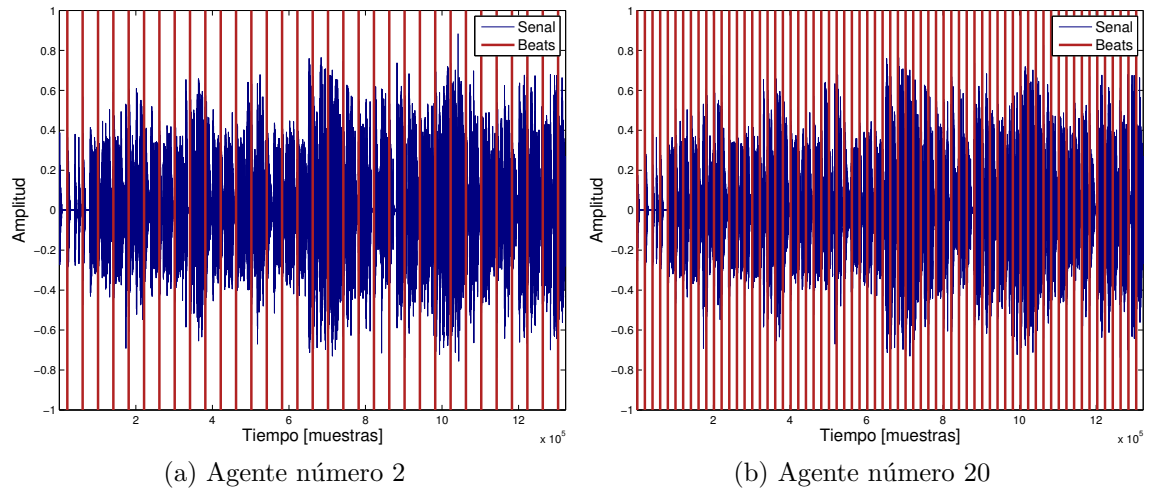


Figura 2.7: Beats detectados

Performance agente 2:

Cont-Based: cC: 00.00 cT: 00.00 aC: 100.00 aT: 100.00  
 F-Mesure: f: 67.47 p: 100.00 r: 50.91 a: 50.91

---

Se puede ver que algunas medidas de desempeño penalizan fuertemente este tipo de errores mientras que otras los admiten de distintas formas y lo penalizan en mayor o menor medida.

Por último si se elije como mejor agente al número 1 la performance alcanzada será perfecta:

Performance agente 1:

Cont-Based: cC: 100.00 cT: 100.00 aC: 100.00 aT: 100.00  
 F-Mesure: f: 100.00 p: 100.00 r: 100.00 a: 100.00

---

En este caso la performance del algoritmo hubiera sido perfecta.

## 2.4. conclusiones

anda en general muy rico

qedaron algunas dudas de implementacion

se puede mejorar la parte de la eleccion del agente ganador

las debilidades pueden estar en la dificultad de elegir el agente posta... nos pasa a veces q agarramos multiplos... como en la sección 2.3.2. Se puede pensar en mejorarlo para elegir el correcto aunque será una difícil tarea el pasaje a lo semántico

en general esta bastante robusto, no se usan umbrales duros ni cosas demasiado arbitrarias

es un sistema muy bien pensado donde se favorece a los mejores y la competencia parece ser un buen camino

tiene una buena relación entre la inercia y la rapidez de respuesta

en silencios de una canción sigue marcando el beat como venia por un rato..

buenos resultados auditivamente hablando



---

## Bibliografía

---

- [1] João Lobato Oliveira, Fabien Gouyon, Luis Gustavo Martins, Luis Paulo Reis, IBT: A real time tempo and beat tracking system, In *11th International Society for Music Information Retrieval Conference, ISMIR*, 2010.
- [2] S. Dixon. Automatic extraction of tempo and beat from expressive performances. In *Journal of New Music Research*, 30(1):39–58, 2001.
- [3] S. Dixon. Onset detection revisited. In *in Proceedings of the 9th International Conference on Digital Audio Effects*, pages 133–13, Montreal, Canada, 2006.
- [4] F. Gouyon, P. Herrera, and P. Cano. Pulse-dependent analyses of percussive music. In *AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio*, 2002.