ALGORITMO PARA LA DETECCIÓN EN TIEMPO REAL DEL TONO EN SEÑALES MUSICALES MONOFÓNICAS POR SEPARACIÓN-ACUMULACIÓN ARMÓNICA (SAA)

PACS: 43.66.Hg

Jaime Serquera Peyró [1]; Juan Luís Corral González [2].
Escuela Politécnica Superior de Gandía.
Avda. de Alicante, 17, 5º.
46700, Gandía (Valencia)
España
Tel: 34 96 286 50 73
E-Mail: jaiserpe@epsg.upv.es (1) ; jlcorral@dcom.upv.es (2)

ABSTRACT

With the goals of real-time feasibility and low delay, mandatory of any 'on-stage' application, this paper describes an approach for the analysis of monophonic musical signals, capable to estimate the pitch in the way of notes along the time for many real-life situations and musical setups.

RESUMEN

La presente comunicación aborda el problema de la detección del tono en señales musicales monofónicas 'tonales', en términos de notas a lo largo del tiempo, mediante técnicas de coste computacional reducido como para poder plantear sistemas en tiempo real que sean útiles en cualquier aplicación musical en vivo.

I. INTRODUCCIÓN

Al hablar de **señales musicales monofónicas** entenderemos señales en las que en cada instante encontramos una única nota musical. Cuando varias notas alcanzan el oído mezcladas en una sola señal acústica, estamos ante el caso de una señal polifónica, que puede ser producida por un solo instrumento (polifónico, como el piano o la guitarra) o por varios instrumentos.

Al hablar de **señales musicales tonales** entenderemos señales que son el resultado de instrumentos musicales preferentemente acústicos tales que producen una sensación precisa y absoluta de tono o afinación, lo cual se debe a que la vibración que generan para producir cada nota concentra mayoritariamente su energía en posiciones espectrales discretas y armónicas entre sí (una trompeta, un piano o una guitarra). Este requisito excluye instrumentos cuyo sonido es claramente ruidoso al carecer de parciales diferenciados (como los platos, caja y otros), así como instrumentos en los que los parciales, pese a existir, no guardan relaciones armónicas entre ellos y que dan lugar a una confusa o imprecisa sensación de tono (por ejemplo congas, timbales de batería, etc.). Los instrumentos excluidos por esta consideración son generalmente percusivos.

Los conversores Tono-MIDI son dispositivos a través de los cuales instrumentos musicales convencionales pueden ser utilizados como controladores de sintetizadores digitales. El sonido del instrumento musical es analizado para determinar las notas que ha producido y generar unos mensajes MIDI (protocolo estándar de comunicación entre sintetizadores) que serán transmitidos al sintetizador para que reproduzca las mismas notas. En esta aplicación, la detección del Tono ha de realizarse en **tiempo real**.

El sonido de los instrumentos musicales acústicos tonales proviene de la vibración periódica de algún elemento (cuerda en el violín, columna de aire en la flauta, etc.). Desde el punto de vista de la señal resultante, el análisis de Fourier de esa señal periódica manifiesta que la energía de la señal producida por el instrumento se concentra en un conjunto discretro de frecuencias {fo,2fo,3fo,...nfo} conocido con el nombre de **serie armónica**, relacionadas entre sí de una forma muy sencilla: son todas múltiplos enteros de una única frecuencia, la frecuencia fundamental.

Las composiciones propias de la corriente musical europea se desarrollan recorriendo notas de una 'escala' (conjunto discreto de frecuencias distribuidas de manera uniforme en un eje log-frecuencial), la escala temperada.

Hay que resaltar el hecho de que las frecuencias de los primeros parciales de la serie armónica de un sonido, al ser múltiplos enteros de su fundamental, coinciden con bastante exactitud con las frecuencias fundamentales de ciertas notas de la escala temperada. Así, podemos construir la serie armónica de un sonido, representando cada armónico con la nota de la escala temperada cuya frecuencia fundamental coincida con la frecuencia de dicho parcial. Es necesario tener presente esta particularidad, ya que será idea fundamental en la estrategia para la detección de tono del algoritmo que se propone en este estudio.

Conseguir un procedimiento automático de análisis en tiempo real, de una señal musical en términos de notas a lo largo del tiempo, es nuestro objetivo principal.

II. ALGORITMO SAA

La solución que se propone al problema de la detección de Tono es un algoritmo que opera en el dominio de la frecuencia y la estrategia que se ha empleado se basa en la *separación* y la *acumulación armónica* del espectro de una forma conjunta y directa (sin tener que realizar una transformación de la señal). Posteriormente, se lleva a cabo una reordenación, según la teoría de la serie armónica, de las distintas partes del espectro separadas, en busca de la nota musical que le corresponda.

El algoritmo está pensado para calcular directamente el nombre de la nota musical (no se busca el valor de la frecuencia fundamental del sonido). El nombre de los sonidos musicales se compone de dos partes:

-*Nombre de nota*: Do, Do#, Re, Re#, Mi, Fa, Fa#, Sol, Sol#, La, La# y Si.

-*Nº de octava*: 1, 2, 3, 4, 5, 6. (son las octavas en las que trabaja el algoritmo, concretamente del Do#1 al Do6, por tanto cubre un rango de 5 octavas).

Basándonos en esto, el proceso de cálculo se divide en dos etapas. Una para hallar el *Nombre de nota*, y la otra para hallar el *Nº de octava* (a partir del resultado de la primera etapa).

1ª ETAPA: Cálculo del *Nombre de nota.*

Antes de comenzar el procesado contamos con 12 señales, una para cada *Nombre de nota* (Sdo, Sdo#, Sre, Sre#, Smi...) guardadas en memoria, las cuales multiplicaremos con cada trama de la señal de audio (convolución en frecuencia).

Formación de las señales:

En el dominio frecuencial, cada una de las 12 señales está formada por bandas cubriendo las frecuencias que correspondan con las frecuencias fundamentales del *Nombre de nota* en cuestión en sus diferentes octavas.

Las señales se construyen en el dominio temporal y las bandas se crean con Sincs elevados al cuadrado y desplazados en frecuencia (Figura 2). Por ejemplo, la señal correspondiente al *Nombre de nota* 'si' se obtendrá según la expresión:

$$S_{si}(t) = \sum_{n=1}^{n=6} sinc^2\left(\frac{BWsi_n}{2} \cdot t\right) \cdot e^{-i\,2p\ fsi_n\ t}$$

donde *n* es el nº de octava, *BW* es el ancho de banda del armónico fundamental de la nota s*i*n, y *f* es la frecuencia fundamental de la nota s*i*n.

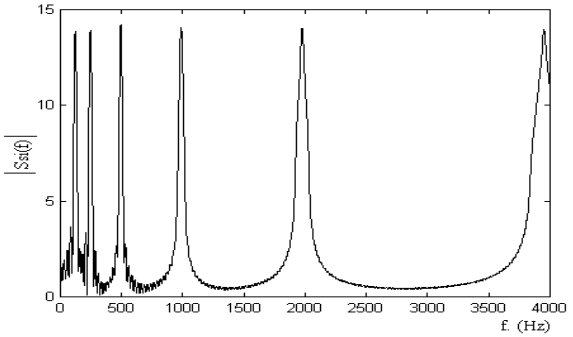


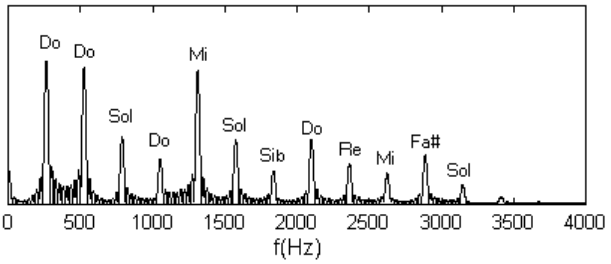
Figura 2: Representación espectral de la señal Ssi

Procesado:

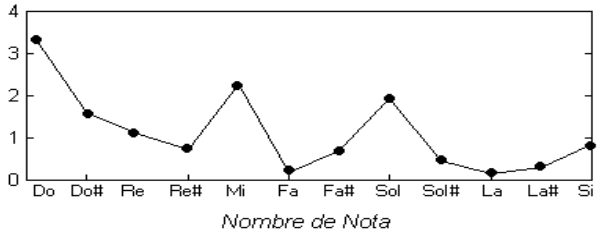
1) Multiplicamos cada una de las 12 señales con la trama de audio. En el dominio frecuencial estamos convolucionando, por lo tanto, cada convolución significa el desplazamiento a la frecuencia cero de las partes espectrales del sonido que les corresponde un mismo *Nombre de nota.* Es aquí donde realizamos ya la acumulación de la energía del espectro.

2) Calculamos el peso de la frecuencia cero en cada una de las 12 señales resultantes de las multiplicación. Con esto conseguimos una representación del espectro del sonido de forma que sabemos cuál es la energía que los armónicos aportan a cada *Nombre de nota*. El peso (índice de energía) se determina calculando el valor absoluto al cuadrado del sumatorio de cada una de las 12 señales.

$$P_{si} = \left|\sum S_{si}[n] \cdot X[n]\right|^2$$



a)



b)

Figura 3: ***a)*** *Espectro correspondiente a una trama de un sonido de saxo ejecutando la nota DO3.*
b) *Representación de los 12 pesos correspondientes a esta misma trama de audio*

3) Determinamos el más significativo de los 12 pesos, que no el máximo. El objetivo es encontrar un peso que corresponda inequívocamente con un armónico del sonido, para ello buscaremos el peso que más diferencia de amplitud tenga con respecto a sus pesos vecinos. Esto se hace para evitar errores producidos por el efecto del enventanado cuando se trabaja en el dominio frecuencial (en frecuencias bajas hay solapamiento entre las bandas de *Nombres de nota* contiguos).

4) Por último faltaría saber a qué serie corresponde este armónico. Para ello, basándonos en la teoría de la serie armónica, haremos cuatro combinaciones (en el algoritmo se consideran cuatro posibilidades) a partir de los 12 pesos calculados anteriormente (el algoritmo funciona correctamente si se tienen en cuenta cuatro pesos en cada combinación).

Ej: Si el peso más significativo es el del Do:

- 1ª combinación, para ver si corresponde al armónico fundamental de la serie armónica de la nota DO:

PSol + PMi + PLa# + Pre

- 2ª combinación, para ver si corresponde al 3º armónico de la serie armónica de la nota Fa:

PFa + PLa + PRe# + PSol

- 3ª combinación, para ver si corresponde al 5º armónico de la serie armónica de la nota Sol#:

PSol# + PRe# + PFa# + PLa#

- 4ª combinación, para ver si corresponde al 7º armónico de la serie armónica de la nota Re:

PRe + PLa + PFa# + PMi

Calculamos el máximo de estas 4 combinaciones y sabremos a qué armónico y por tanto, a qué serie corresponde el peso más significativo (con lo cual sabremos también el *Nombre de nota* del sonido).

Si el máximo de estas 4 combinaciones no supera un umbral (establecido de forma empírica y proporcional al peso más significativo), significa que el peso más significativo es directamente el *Nombre de nota*; esto ocurre cuando el sonido es muy puro, es decir, cuando el nivel del armónico fundamental es muy superior al de los restantes.

2ª ETAPA: Cálculo del *nº de octava.*

Una vez conocido el *Nombre de nota*, tenemos que determinar el *nº de octava.* Del mismo modo que en la 1ª Etapa, ahora también necesitamos crear unas señales que permanecerán en memoria. En este caso será una señal para cada nota musical, con una banda en su frecuencia correspondiente. Por ejemplo, la señal correspondiente a la nota *n* se obtendrá según la expresión:

$$S_n(t) = sinc^2\left(\frac{BWn}{2}\cdot t\right)\cdot e^{-i\,2p\,f_n\,t}$$

donde *n* es la nota musical en cuestión, *BW* es el ancho de banda del armónico fundamental de la nota *n*, y *f* es la frecuencia fundamental de la nota *n*

Procesado:

Multiplicamos la trama de audio por las señales *Sn* que posean el mismo *Nombre de nota* calculado en la 1ª Etapa y en sus distintas octavas. Calculamos el peso de f = 0 para cada una de las señales resultantes de la multiplicación, de la misma forma que en la 1ª Etapa. Como el algoritmo trabaja en el rango de 5 octavas, en esta 2ª Etapa se calcularán un total de 5 pesos.

$$P_n = \left|\sum S_n[n]\cdot X[n]\right|^2$$

Partiendo de la octava más baja, buscamos el primer peso significativo, es decir, aquel que supere un umbral (establecido de forma empírica y que diferencia entre la presencia de un armónico o la presencia de ruido).

III. RESULTADOS

Una serie de parámetros determinan la capacidad de un algoritmo de detección de Tono para ser utilizado en tiempo real. El **tiempo de respuesta** es el retardo existente entre el instante en que una nota es ejecutada por un instrumento y el instante en que el sintetizador empieza a generar el correspondiente sonido. Idealmente, el tiempo de respuesta debe ser lo suficientemente corto como para que el ejecutante del instrumento no perciba el retardo. El sistema auditivo humano tiene un tiempo de respuesta medio de unos 50 ms., es decir, dos señales acústicas que llegan con un desfase temporal inferior a 50 ms., se perciben como una señal única (Llin91). Luego este será el límite de tiempo para que el sintetizador genere el sonido sin que el instrumentista perciba el retardo. Considerando que el retardo introducido por el sintetizador es despreciable, el tiempo de respuesta es igual a la suma de la *longitud de la trama* inicial de señal analizado y el *tiempo de cómputo* del algoritmo.

El algoritmo SAA trabaja con una frecuencia de muestreo de 8000 Hz. Se ha elegido la frecuencia de muestreo más baja que pueda registrar la frecuencia fundamental del sonido más agudo (Do7 del flautín). En cuanto a la elección de la longitud de las tramas, nos encontramos ante un conflicto: por una parte, interesa trabajar con longitudes de trama pequeñas para obtener un funcionamiento en tiempo real. Por otro lado, como trabajamos en el dominio frecuencial, existe el problema del enventanado: cuanto más pequeña sea la longitud de las tramas, mayor será la distorsión provocada en el espectro del sonido a analizar. La solución de compromiso que se ha tomado es de 300 puntos. Así pues, las tramas son de 37 ms., por tanto, *el tiempo de cómputo* del algoritmo ha de ser inferior a 13 ms. para que su funcionamiento sea en *tiempo real*.

Una forma de evaluar el *tiempo de cómputo* es el cálculo de los ciclos de reloj que un procesador DSP requeriría para ejecutar el algoritmo.

El algoritmo propuesto en este estudio ocupa la mayor parte de su tiempo de cómputo en el cálculo de los Pesos, tanto en la 1º como en la 2ª Etapa. Se considera que las

instrucciones para implementar los bucles, detección de máximos, etc. requieren una cantidad de tiempo despreciable con respecto al cálculo de los Pesos.

Recordemos que la expresión para calcular los pesos era:

$$P = \left| \sum S[n] \cdot X[n] \right|^2$$

donde: S[n] es la señal de Sincs desplazados (señal compleja) con una longitud de 300 puntos, X[n] es la trama de audio (señal real) con una longitud de 300 puntos

Así, esta ecuación implica 2L multiplicaciones, 2L sumas y el valor absoluto al cuadrado que conlleva 2 multiplicaciones y 1 suma (siendo L=300 la longitud de las tramas).

Asumimos que el procesador DSP es capaz de realizar las operaciones de multiplicación y suma en un único ciclo de reloj. Un ejemplo de este tipo de procesador es el ADSP-2100.

Por tanto, cada Peso necesita (2L+2L+2+1) = 1203 ciclos de reloj. Si en la 1ª Etapa del algoritmo se calculan 12 Pesos (correspondientes a cada una de las 12 notas musicales) y en la 2ª Etapa se calculan 5 Pesos (correspondientes a las 5 octavas en las que trabaja el algoritmo) tenemos un total de 1203*12+1203 *5 = 20451 ciclos de reloj.

A modo de comparativa, uno de los últimos algoritmos para la detección de Tono en tiempo real propuesto por A. Choi en 1996: “Real-Time Fundamental Frequency Estimation by Least-Square Fitting” (Cho97), requiere un total de 398700 ciclos de reloj.,

Con la función ‘tic-toc’ de Matlab se puede calcular el tiempo de cómputo del algoritmo. Como ejemplo, se puede decir que el algoritmo tarda 0.16 segundos en procesar (en un Pentium 350) un sonido de 1.216 segundos de duración, es decir 5 ms. por trama, con lo cual el funcionamiento en tiempo real está conseguido.

El funcionamiento del algoritmo SAA se ha contrastado con otro algoritmo basado en la técnica SFFT (Pis79). En cuanto a la exactitud de resultados, el algoritmo SFFT provoca menos errores esporádicos, pero por el contrario, produce numerosos errores de octavación que no se dan en el algoritmo SAA (Figura 4). En cuanto al tiempo de respuesta, el algoritmo SAA trabaja a más del doble de velocidad.

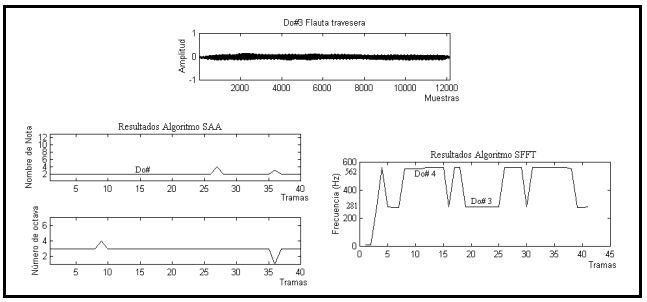


Figura 4

A continuación se enumeran las principales limitaciones del algoritmo SAA:

- Errores esporádicos (ver Figura 4), lo cual implica facilidad de corrección.
- Ofrece malos resultados ante fuertes resonancias (las cuales aparecen sobretodo en las notas más graves de cada instrumento). Este es un problema común en casi todos los métodos de detección de tono.
- Resolución en frecuencia de 12 notas/octava: no detecta ornamentos ni modulaciones, tales como glissandos y vibratos.
- No es aplicable a señales con reverberación.

IV. SUMARIO

Se ha presentado un algoritmo para la detección de Tono en señales musicales monofónicas tonales. A diferencia de otros, se ha pretendido en todo momento que el coste computacional fuera lo más reducido posible como para conseguir un correcto funcionamiento en tiempo real.

El procedimiento está basado en la reagrupación armónica del espectro y la originalidad del mismo radica en la forma en que se lleva a cabo esta separación-acumulación de la energía espectral. Atendiendo al objetivo final, que es conseguir una representación de notas a lo largo del tiempo, los distintos armónicos que constituyen el sonido en cuestión son agrupados según el nombre de nota que les correspondería en caso de que fuesen considerados como frecuencias fundamentales.

Así, en una primera etapa la energía del espectro del sonido se desglosará en doce grupos que representan cada una de las doce notas musicales. A partir de aquí, se establece un sistema de identificación del nombre de nota del sonido basado en la teoría de la serie armónica, que consiste, básicamente, en la reconstrucción de varias series armónicas a partir de las energías de las doce agrupaciones realizadas. Aquella serie que consiga englobar mayor energía será la que corresponda a la nota musical del sonido en cuestión.

Una vez conocido el nombre de la nota del sonido, se procede a averiguar el número de octava de la misma. Para ello se realiza un filtrado del sonido quedándonos sólo con las partes del espectro que les corresponde el mismo nombre de nota calculado. A partir de estas señales filtradas se realiza una búsqueda ascendente en frecuencia para averiguar cuál es la primera de ellas que tiene un nivel de energía suficientemente elevado como para considerar que se trata de un armónico del sonido y no se trata de presencia de ruido.

Como son doce las notas musicales, y cinco las octavas que tiene en cuenta el algoritmo SAA, el proceso consiste en escoger una entre doce posibilidades en la primera etapa de cálculo, y otra entre cinco posibilidades en la segunda etapa. De esta forma, podemos ver claramente que el coste computacional es mucho menor que en aquellos algoritmos que tratan de determinar una frecuencia (la frecuencia fundamental del sonido) entre miles de que frecuencias discretas posibles, con el mismo objetivo final que es el de averiguar la nota musical ejecutada.

VI. REFERENCIAS

Cho97 A. Choi, “RealTime Fundamental Frequency Estimation by Least-Square Fitting”, IEEE Trans. Speech & Audio Processing Mar’97 5 (2) pp 201-205

Llin91 J. Llinares, A. Llopis, J. Sancho, “Acústica Arquitectónica y Urbanística”, Servicio de Publicaciones Universidad Politécnica de Valencia, SPUPV-91.640, I.S.B.N. 84-7721-133-7

Pis79 Piszczalski, M. And B. A. Galler. 1979. “Predicting Musical Pitch from Component Frequency Ratios” Journal of the Acoustical Society of America 66(3):710-720