

# FORMATOS DE COMPRESIÓN SIN PÉRDIDAS

LUCAS G. SCHAMUN<sup>1</sup>, ENRIQUE A. SUELDO<sup>1</sup> y NAHUEL A. DE LA VEGA<sup>1</sup>

<sup>1</sup>Estudiante de Ingeniería Electrónica, Facultad Regional Córdoba, Universidad Tecnológica Nacional, (FRC, UTN),  
Maestro López Esq. Cruz Roja Argentina. CP X5016ZAA. Córdoba, República Argentina.  
lgschamun@gmail.com, enriquesueldo94@gmail.com, nahuelsk8.delavega@gmail.com

**Resumen** – *La utilización de los formatos de compresión sin pérdidas se debe a que la calidad de la señal no es alterada, se trata de algoritmos útiles en aplicaciones de alta fidelidad. El factor de compresión es más dependiente del género musical, que el códec utilizado. Los tiempos de decodificación son menores que los de codificación. En este trabajo se realiza un análisis comparativo de la compresión y demora de procesamiento para los distintos formatos de compresión sin pérdidas.*

**Abstract** – *Lossless compression formats are used because the quality of the signal is not altered, it is about of useful algorithms in high-fidelity applications. The compression factor is more dependent on the musical genre, than the codec used. Decoding times are less than coding times. In this work, a comparative analysis of the compression and processing delay is performed for different compression formats without losses.*

## 1. INTRODUCCIÓN

La compresión de audio es la reducción del tamaño de los archivos de audio. Los algoritmos utilizados normalmente son llamados códecs.

Los formatos sin compresión presentan datos PCM (modulación de códigos de pulsos), estos contienen toda la información que salió del convertidor analógico a digital, sin ninguna omisión y por eso, tienen la mejor calidad. (Wikipedia, 2017)

Existen dos tipos de compresión, basados en algoritmo de compresión sin pérdida o algoritmo de compresión con pérdida. [4]

Estos dos métodos se basan en que las señales de audio poseen un grado considerable de redundancia, usando la estrategia de modelización de la señal que permite predecirla a partir de algunos datos anteriores.

Los formatos con pérdida utilizan criterios psicoacústicos para descartar información que en condiciones normales resultaría inaudible para la mayoría de las personas.

Los formatos de compresión sin pérdida se basan en consideraciones estadísticas y teoría de la información, permitiendo una reconstrucción perfecta de la señal. Son muy utilizados para almacenamiento a largo plazo, repositorios en Internet de música, bases de sonidos y transmisión por redes.

El objetivo de este informe técnico es realizar un análisis comparativo de los distintos CSP (codificadores sin pérdidas), resumiendo la información de diferentes fuentes. Estas analizan el rendimiento de los CSP para diversas piezas musicales, en base de la compresión alcanzada y la demora de procesamiento, tanto para la codificación como para la decodificación.

Este trabajo se encuentra organizado de la siguiente manera. En la sección 2 se realizó una descripción de los conceptos generales de la codificación sin pérdidas. En la sección 3 se encuentran dos análisis comparativos de codificadores de audio. Luego en la sección 4 se realizó una comparación de los diferentes algoritmos de compresión.

## 2. CODIFICACIÓN SIN PÉRDIDAS

Al codificar una señal de audio con este método, el número de muestras luego de descomprimir la señal es la misma.

La compresión sin pérdidas se basa en eliminar la redundancia de la señal, esta es una propiedad de los mensajes, que consiste en tener partes predecibles a partir del resto del mensaje y que por tanto en sí mismo no aportan nueva información o repiten parte de la misma. En resumen, si la señal tiene patrones repetitivos, ésta es redundante y por lo tanto fácil de predecir. [2]

Podemos describir las siguientes fuentes de redundancia:

- Intercanal: es la redundancia entre canales, para sistemas estereos o multicanal, las señales de cada canal pueden ser muy parecidas, y se puede eliminar.
- Intracanal: redundancia temporal manifestada en una correlación entre muestras continuas, se realiza una predicción a partir de una combinación lineal de muestras adyacentes en el espacio o en el tiempo.
- Estadística: cuando en la señal predominan unos pocos valores que se presentan con mucha probabilidad, frente a otros de baja, se utilizan códigos de longitud variable, en donde con pocos bits son

codificados los símbolos probables y con cadenas de bits más largos los símbolos menos probables.

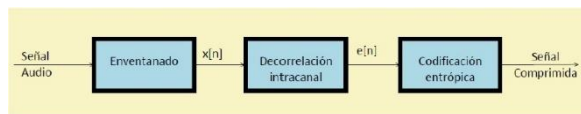


Figura 1: Esquema general de los compresores de audio sin pérdidas.

### 2.1 Enventando

La señal de entrada, en primer lugar, es acotada por tramos mediante un enventanado, si no se divide la señal, es un inconveniente en cuanto a tiempo y computacional se refiere.

### 2.2 Decorrelación intracanal

La etapa siguiente es la decorrelación intracanal, está basada en la eliminación de la redundancia temporal estimando eficientemente la señal de entrada. Gracias a esta estimación resulta un error de aproximación o residuo,  $e[n]$  (diferencia entre la muestra original y su valor aproximado) con menor potencia que la entrada, de esta manera el conjunto de datos es más fácilmente comprimible, es decir un espectro más plano, por lo tanto, se podrá codificar con menos bits.

Se puede estimar vía predicción lineal o extrayendo unos pocos coeficientes de la trasformada de la entrada en el dominio frecuencial.

En este caso se utiliza el método del modelo predictivo, donde cada muestra de la entrada  $x(n)$  es aproximada por la combinación lineal de  $N$  muestras de entrada anteriores (FIR) y  $M$  muestras de salidas anteriores (IIR). Como se puede observar en la Figura 2.[3]

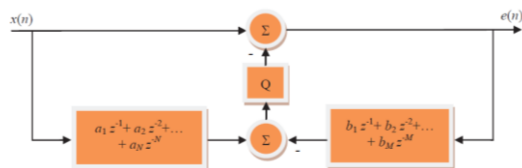


Figura 2: Diagrama de bloques de la sección de decorrelación.

### 2.3 Codificación entrópica

Codificar cada una de las muestras de la señal anterior de la manera más eficiente posible, se denomina codificación de entropía. Su principio básico es que se asignan códigos más cortos a valores con mayor probabilidad de aparición.

En el receptor, el decodificador reconstruye la señal original de la siguiente manera. Los datos entrantes se demultiplexan, luego se extrae y decodifica el residuo, y se lo añade a los valores estimados por el predictor.

### 2.4 Parámetros de los CSP

– Factor de compresión (FC): se define como la razón entre la longitud de la señal original y la longitud de la señal comprimida.

– Rate (R): se define como cociente entre el tiempo de reproducción del archivo original y el tiempo de codificación o decodificación según corresponda. Es una medida de la velocidad de procesamiento del codificador o decodificador. (Marengo, et al; 2011).

## 2.5 Ejemplos de CPS

La Tabla 1 muestra ejemplos de CPS.

Codecs
ALAC (Apple Lossless)
DST (Direct Stream Transfer)
FLAC (Free Lossless Audio Codec)
LA (Lossless Audio)
LPAC (Lossless Predictive Audio Codec)
LTAC (Lossless Transform Audio Codec)
MLP (Meridial Lossless Packing)
Monkey's Audio (APE)
MPEG-4 ALS
MPEG-4 SLS
OptimFROG
QDesign
RealAudio Lossless
RKAU
Shorten (SHN)
TTA (True Audio)
WavPack
WMA lossless (Windows Media Audio Lossless)

Tabla 1: Ejemplos de tipos de codificadores sin pérdidas.

## 3. ANÁLISIS COMPARATIVO DE CODIFICADORES DE AUDIO

Utilizando el análisis realizado por [3] y [9], se analizó el rendimiento de los CSP para diversas piezas musicales, en base de la compresión alcanzada y la demora de procesamiento, tanto para la codificación como para la decodificación.

### 3.1 Análisis I

En el análisis de [3], se utilizaron entre 15 y 20 archivos de audio para cada uno de los siguientes géneros musicales: folklore argentino, tango, jazz, música clásica, pop y rock. Promediando el FC y el R para todas las piezas de cada género.

En la Tabla 2 se visualizan los códec con sus respectivas versiones, el hardware utilizado se encuentra en la Tabla 3.

Codec
FLAC v.1.2.1
LPAC v.3.08
Monkey's Audio v.4.06
MPEG-4 ALS RM22rev2
OptimFrog v.4.600ex
Shorten v.3.6.1
TAK v.2.0.0
TTA v.3.4.1WavPack a v.4.60.1

Tabla 2: Lista de codecs utilizadas para el análisis I.

	Procesador	RAM y Disco rígido	Sistema operativo
PC1	HP Pavilion DV7 -3065DX / 2.4 GHz AMD Turion II Ultra Dual-Core Mobile M600	2 x 2 GB SDRAM DDR2-800 / Samsung HM321HI 5400 rpm SATA (320 GB)	Windows Vista Home Premium
PC2	Intel Rock Lake D865PERL / 3 GHz Pentium IV HT	2 x 512 MB SDRAM DDR / Maxtor 6Y080L0, Ultra- ATA133 (80 GB)	Windows XP Service Pack 3
PC3	MSI K9N6PGM-F/PI / 2.2 GHz AMD Athlon 64 x 2 Dual Core	2 GB SDRAM DDR2 / WDC WD1600AABS SCSI (149 GB)	Windows XP Service Pack 3

Tabla 3: Computadoras utilizadas para el análisis I.

### 3.1.1 Codificación en modo normal

En este caso se utilizó la computadora PC1. El factor de compresión para los diferentes géneros se muestra en la Figura 4, el promedio en la Tabla 4 y en la Figura 5 una representación gráfica con respecto al Rate.

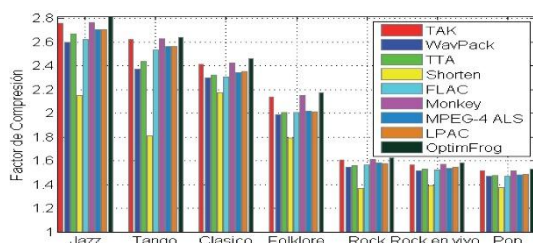


Figura 4: Factor de compresión asociado a cada género musical analizado con los CSP etiquetados en color (codificación en modo normal).

Género	Jazz	Tango	Clásico	Folklore	Rock	Rock en vivo	Pop
FC promedio	2.64	2.46	2.34	2.03	1.57	1.53	1.48

Tabla 4: Factor de compresión promedio de cada género musical.

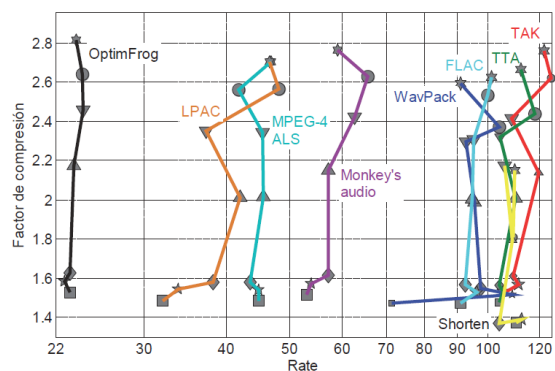


Figura 5: Representación gráfica del rate con respecto al factor de compresión obtenido para cada género musical en modo normal. Los géneros analizados son jazz (\*), tango (●), música clásica (▼), folklore (▲), rock (◆), rock en vivo (★) y pop (■).

Se puede apreciar que el factor de compresión varía de acuerdo al género musical, el mejor resultado de compresión se da para el jazz. Esto se debe a que la señal tiene un mayor grado de redundancia.

El CSP con menor Rate es OptimFrog, mientras que el mayor es el TAK.

### 3.1.2 Codificación en modo rápido y modo máximo

También se realizó el mismo análisis para estos dos modos, esto se puede observar en la Figura 6 y Figura 7. Como es de esperar, el modo máximo presenta un aumento de la velocidad de procesamiento en la computadora PC1, con respecto al modo máximo.

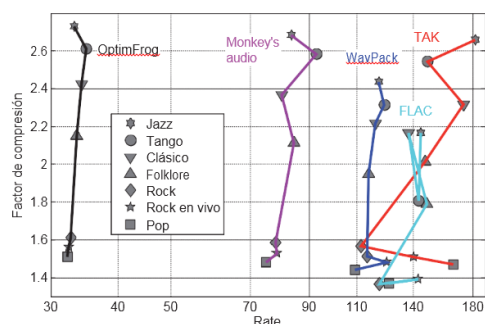


Figura 6: Representación gráfica del rate con respecto al factor de compresión obtenido para cada género musical en modo rápido.

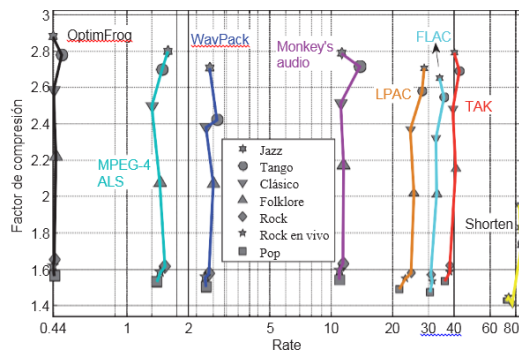


Figura 7: Representación gráfica del rate y factor de compresión obtenidos para cada género musical con los CSP mencionados en la Tabla 1 en modo máximo.

### 3.1.3 Decodificación

En este caso requiere menor tiempo, especialmente para los modos normal y máximo, cabe aclarar que para Shorten y Monkey's Audio lo comentado no es válido, porque utilizan un algoritmo simétrico para la codificación y decodificación.

La Tabla 5 muestra los tiempos de codificación / decodificación para una sola pieza codificada en los modos rápido y máximo (sólo en los codecs donde fue posible). Como puede observarse, TAK, FLAC y WavPack son los codificadores más rápidos. Esto los hace útiles para la reproducción de archivos codificados en dispositivos de audio de bajo costo o portátiles.

Codec	Modo máximo	Modo rápido
TAK	40 / 48	181 / 59
WavPack	3 / 51	122 / 58
TTA	119 / 48 (único modo)	
Shorten	82 / 56	115 / 25 (modo normal)
FLAC	32 / 41	124 / 56
Monkey's Audio	11/4	79 / 32
MPEG-4 ALS	1 / 9	46 / 35 (modo normal)
LPAC	24 / 32	87 / 35
OptimFrog	0.44 / 1	33 / 30

Tabla 5: Tiempos de codificación y decodificación en PC0 de la pieza "Si te vas", interpretada por A. Zitarrosa.

### 3.1.4 Codificación y decodificación para diferente hardware

La compresión de los distintos códecs, no difieren en gran medida con respecto a las computadoras utilizadas en la Tabla 3.

Para los modos rápido, normal y máximo la región de operación es similar. El rate debe ser escalado a los límites del hardware.

## 3.2 Análisis II

Para este análisis se utilizó una computadora Intel Core2Duo T9600 con 4GB de RAM, usando Ubuntu 12.10.(Xiph, 2001-2004)

Las señales de audio utilizadas se componen por 29 álbumes de CDs, cuyos géneros son: rock, pop, metal, clásico de todo el mundo.

En la Figura 8 y Figura 9 se muestran los gráficos de codificación y decodificación de todos los archivos de audio, los mismos resultaron de un promedio de cada archivo utilizado.

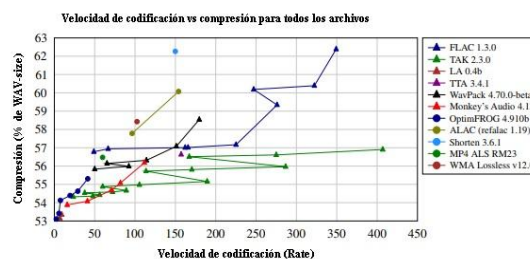


Figura 8: Velocidad de codificación vs. compresión.

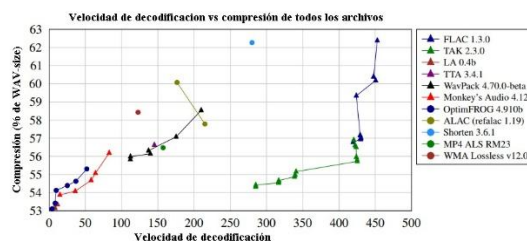


Figura 9: Velocidad de decodificación vs. compresión.

El formato TAK 2.3.0 es el códec que alcanza la mayor velocidad de compresión, en cambio OptimFrog 4.910b es el más lento, tal como resultó en el análisis I.

El códec FLAC 1.3.0 es el que alcanza la mayor compresión, tanto en codificación, como en decodificación. Además, posee la mayor velocidad de decodificación.

## 4. COMPARACIÓN DE ALGORITMOS DE COMPRESIÓN

Los algoritmos utilizados para la compresión de audio son los siguientes:

- Compresión Huffman: es un algoritmo usado para compresión de datos. El término se refiere al uso de una tabla de códigos de longitud variable para codificar un determinado símbolo (como puede ser un carácter en un archivo), donde la tabla ha sido rellena de una manera específica basándose en la probabilidad estimada de aparición de cada posible valor de dicho símbolo.(Wikipedia, 2018).
- Compresión Shannon Fano: es una técnica para construir un código prefijo basado en un conjunto de símbolos y sus probabilidades (estimadas o medidas). No es óptimo en el sentido de que no consigue la menor longitud de palabra código esperado posible como en la codificación Huffman. (Wikipedia, 2015).
- Compresión LZW: la clave del método LZW reside en que es posible crear sobre la marcha, de manera

automática y en una única pasada un diccionario de cadenas que se encuentren dentro del texto a comprimir mientras al mismo tiempo se procede a su codificación. Dicho diccionario no es transmitido con el texto comprimido, puesto que el descompresor puede reconstruirlo usando la misma lógica con que lo hace el compresor y, si está codificado correctamente, tendrá exactamente las mismas cadenas que el diccionario del compresor tenía. (Wikipedia, 2018).

En la Figura 10 y Figura 11 se encuentra una comparación de la compresión y tiempo de compresión para los distintos algoritmos mencionados, cabe destacar que los mismos no son solo utilizados para archivos de audio, sino que también, para distintos tipos de archivo. [1]

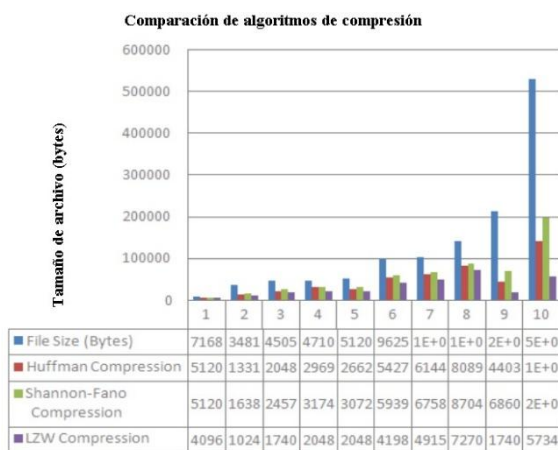


Figura 10: Comparación de algoritmos de compresión vs. tamaño del archivo.

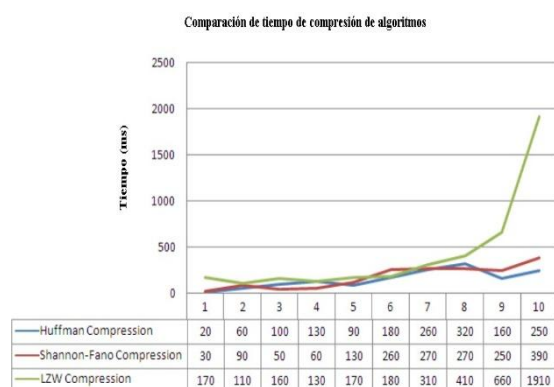


Figura 11: Comparación de tiempo de compresión vs. tamaño del archivo.

## 5. CONCLUSIONES

El factor de compresión es más dependiente del género musical, que el códec utilizado.

En relación al Rate, el de menor es OptimFrog, mientras que los mayores son el TAK y FLAC, de acuerdo al análisis I y II.

Con respecto a los modos de compresión, el modo rápido es conveniente si se desea minimizar el tiempo de procesamiento, a costa de un mayor almacenamiento de la señal y en el modo máximo se utiliza si se desea una mayor compresión.

En general, los tiempos de decodificación son menores que los de codificación, excepto Monkey's Audio.

Para la codificación y decodificación utilizando diferentes computadoras, se notó que la compresión de los distintos códecs no difiere en gran medida.

El algoritmo de compresión sin pérdidas más óptimo es LZW.

## 6. REFERENCIAS

- [1] Bhattacharjee Arup, et al., "Comparison study of lossless data compression algorithms for text data". IOSR Journal of computer engineering, volume 11, pp. 1-5. May-Jun 2013.
- [2] Garcia Sonia, "Implementacion de un codificador de audio sin perdidas". Proyecto fin de Carrera, Universidad Carlos III de Madrid.
- [3] Marengo Rodriguez et al., "Analisis comparativo de codificadores de audio sin perdidas y una herramienta grafica para su seleccion prediccion de su desempeño". Mecánica computacional, volumen XXX, pp. 1-10. Rosario, Argentina, November 2011.
- [4] Miyagra Federico., "Mediciones acústicas basadas en software". Asociacion de acústicos argentinos, volumen I. Argentina, July 2013.
- [5] Wikipedia, Online encyclopedia. *Worst-case execution time*. [https://es.wikipedia.org/wiki/Audio\\_digital](https://es.wikipedia.org/wiki/Audio_digital) case execution time. Revision 11:32, August 28th, 2017.
- [6] Wikipedia, Online encyclopedia., *Worst-case execution time*. [https://es.wikipedia.org/wiki/Codificaci%C3%B3n\\_Huffman](https://es.wikipedia.org/wiki/Codificaci%C3%B3n_Huffman) case execution time. Revision 13:02, May 15th, 2018.
- [7] Wikipedia, Online encyclopedia., *Worst-case execution time*. [https://es.wikipedia.org/wiki/Codificacion\\_Shannon-Fano](https://es.wikipedia.org/wiki/Codificacion_Shannon-Fano) case execution time. Revision 16:33, Jul 27th, 2015.
- [8] Wikipedia, Online encyclopedia., *Worst-case execution time*. <https://es.wikipedia.org/wiki/LZW> case execution time. Revision 13:35, May 15th, 2018.
- [9] Xiph, Org. Foundation., *Worst-case execution time*. <https://xiph.org/flac/comparison.html> case execution time. 2001-2014.

## 7. DATOS BIOGRAFICOS

**Schamun Lucas G.**, nacido en Córdoba el 10/03/1994. Estudiante de ingeniería en electrónica, Universidad Tecnológica Nacional, Facultad Regional Córdoba. Argentina. E-mail: lgschamun@gmail.com.

**Sueldo N. Enrique Alberto**, nacido en Córdoba el 09/08/1994. Estudiante de ingeniería en electrónica, Universidad Tecnológica Nacional, Facultad Regional Córdoba. Argentina. E-mail: enriquesueldo94@gmail.com

**De la Vega Nahuel A.**, nacido en Córdoba el 20/09/1992. Estudiante de ingeniería en electrónica, Universidad Tecnológica Nacional, Facultad Regional Córdoba. Argentina. E-mail: nahuelsk8.delavega@gmail.com