



UNIVERSIDAD NACIONAL  
AUTÓNOMA DE  
MÉXICO

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

---

**PROGRAMA DE MAESTRÍA Y DOCTORADO EN  
INGENIERÍA**

**Programa interactivo para el diseño asistido por  
computadora de instrumentos musicales de viento**

**T E S I S**

Que para obtener el título de:  
**MAESTRO EN INGENIERÍA ELÉCTRICA**  
INSTRUMENTACIÓN ACÚSTICA

**P R E S E N T A**  
**HÉCTOR ALFONSO CORDOURIER MARURI**

**TUTOR**  
**DR. FELIPE ORDUÑA BUSTAMANTE**



*2006*



## *Jurado asignado*

**PRESIDENTE: DR. RICARDO RUÍZ BOULLOSA.**

**SECRETARIO: DR. LUIS A. ÁLVAREZ ICAZA.**

**VOCAL: DR. FELIPE ORDUÑA BUSTAMANTE.**

**1<sup>er</sup> SUPLENTE: MTRA. MARÍA DIEZ-CANEDO FLORES.**

**2° SUPLENTE: DR. PABLO RENDÓN GARRIDO.**

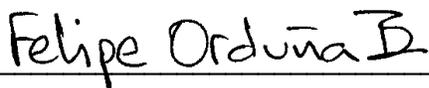
**LUGAR DONDE SE REALIZÓ LA TESIS:**

**CENTRO DE CIENCIAS APLICADAS Y DESARROLLO TECNOLÓGICO, UNAM, CIUDAD UNIVERSITARIA.**

**CIUDAD DE MÉXICO, DISTRITO FEDERAL, MÉXICO.**

**TUTOR DE TESIS:**

**DR. FELIPE ORDUÑA BUSTAMANTE**

  
\_\_\_\_\_

**FIRMA**



# *Agradecimientos*

- A mi mamá, María Teresa Maruri Camacho; mi papá, Alfonso Cordourier Maruri; y mi hermano, Guillermo Cordourier Maruri, por su apoyo y aliento incondicionales.
- A mi tutor, el Dr. Felipe Orduña Bustamante, por su excelente guía y por su gran generosidad.
- A mis sinodales, por su interés y asistencia en el desarrollo de este trabajo.
- A mis profesores de la maestría: Dr. Felipe, M. Antonio, M. Santiago, Dra. Graciela, Dra. Rosalba y Dr. Pablo, por su gran apoyo académico y moral.
- A mi familia cercana, mis tíos y mis primos del estado de México y de Tlaxcala, por su preocupación y cuidado durante este tiempo lejos del hogar.
- A mis amigos de la maestría: Alejandro, Javier, Paloma, Felipe, Antonio, Vicente, Violeta, César, Vinicio y Odín; por su gran ayuda y por hacer de la maestría una de las mejores experiencias.
- A mis amigos de Mérida: Félix, Mario, Denis, William, Alina y Ángel; por su cercanía y solidaridad, desde tan lejos.
- Al Consejo Nacional de Ciencia y Tecnología (CONACYT), por la confianza depositada en mi desarrollo profesional.

*“Toda autoridad emana del recto proceder.”*

- Andrés Manuel López Obrador.

# Índice

<i>Jurado asignado</i> .....	<i>i</i>
<i>Agradecimientos</i> .....	<i>iii</i>
<i>Índice</i> .....	<i>v</i>
<i>Introducción</i> .....	<i>1</i>
<b>I.1 Modelado de instrumentos musicales de viento</b> .....	<b>1</b>
<b>I.2 Objetivo del proyecto de tesis</b> .....	<b>2</b>
<b>I.3 Estructura del documento de tesis</b> .....	<b>2</b>
<i>Capítulo 1: Antecedentes</i> .....	<i>3</i>
<b>1.1 Simulación de instrumentos de viento</b> .....	<b>3</b>
<b>1.2 Acústica de tubos</b> .....	<b>3</b>
1.2.1 Ondas acústicas planas .....	3
1.2.2 Impedancia acústica específica .....	6
1.2.3 Coeficientes de reflexión y transmisión .....	7
1.2.4 Impedancia y reflexión a la entrada de un tubo .....	7
1.2.5 Impedancia y reflexión en un tubo cilíndrico .....	8
1.2.6 Impedancia en un tubo cónico .....	9
1.2.7 Función de reflexión .....	10
<b>1.3 Modelos físicos de los instrumentos de viento</b> .....	<b>11</b>
1.3.1 Excitador - resonador .....	11
1.3.2 La boquilla: Modelo de McIntyre et al. ....	14
1.3.3 La boquilla: Modelo de Verge et al. ....	15
1.3.4 El tubo resonador .....	18
1.3.5 Matrices de transferencia de cilindros y conos .....	19
1.3.6 Matrices de transferencia de agujeros de digitación .....	20
1.3.7 Función de reflexión del resonador completo .....	22
1.3.8 Adaptación a algoritmos digitales .....	24

<i>Capítulo 2: Desarrollo de la aplicación</i> .....	<b>27</b>
<b>2.1 Estructura del programa</b> .....	<b>27</b>
2.1.1 Requerimientos de la interfaz de usuario.....	27
2.1.2 Estructura de ventanas y bibliotecas.....	27
<b>2.2 Interfaz de usuario</b> .....	<b>29</b>
2.2.1 Ventana de bienvenida.....	29
2.2.2 Ventana de diseño.....	29
2.2.3 Ventanas de creación o edición de elementos .....	31
2.2.4 Ventana de simulación .....	33
2.2.5 Ventanas graficadoras.....	35
2.2.6 Ventana del editor de secuencias .....	36
2.2.7 Ventana de opciones generales.....	37
<b>2.3 Bibliotecas de operación</b> .....	<b>38</b>
2.3.1 Biblioteca de estructura de datos del instrumento (Inst_Saver.cpp).....	38
2.3.2 Biblioteca de objetos gráficos (Inst_Draw.cpp) .....	41
2.3.3 Biblioteca del método de matrices de transferencia (Woodwind.cpp) .....	42
2.3.4 Biblioteca de sonido y manejo de secuencias (Sound_Maker.cpp).....	45
<b>2.4 Algoritmos principales</b> .....	<b>47</b>
2.4.1 Algoritmo de cálculo de la función de reflexión.....	47
2.4.2 Algoritmo de simulación de sonido: McIntyre et al.....	49
2.4.3 Algoritmo de simulación de sonido: Verge et al. ....	51
<b>2.5 Prototipos de otros modelos</b> .....	<b>53</b>
2.5.1 El clarinete.....	54
2.5.2 La cuerda tensa frotada.....	54
 <i>Capítulo 3: Resultados</i> .....	 <b>57</b>
<b>3.1 Pruebas comparativas</b> .....	<b>57</b>
<b>3.2 Discusión de resultados</b> .....	<b>65</b>
 <i>Conclusiones y recomendaciones</i> .....	 <b>67</b>
<b>CR.1 Conclusiones</b> .....	<b>67</b>
<b>CR.2 Recomendaciones para trabajos posteriores</b> .....	<b>68</b>
 <i>Bibliografía</i> .....	 <b>69</b>

# *Introducción*

## ***1.1 Modelado de instrumentos musicales de viento***

La gran mayoría de los instrumentos musicales profesionales de la actualidad presenta un muy alto nivel de perfeccionamiento. Las técnicas de manufactura de estos instrumentos son, generalmente, conocimientos obtenidos por la experimentación empírica a través del paso de muchas generaciones. Estos conocimientos han permitido la fabricación de instrumentos de gran calidad desde las épocas medievales.

En contraste, el estudio científico de los instrumentos musicales, que tiene por objeto determinar las causas y los procesos que les brindan su sonoridad característica, es una investigación relativamente reciente. Más reciente aún, es la propuesta de utilizar estos estudios para generar herramientas de ingeniería que permitan evaluar los instrumentos, mejorar su sonido, simplificar su construcción o interpretación, o crear nuevos diseños.

Para poder avanzar en esta propuesta, es preciso que los resultados físicos de las investigaciones sean modelados matemáticamente, para poder ser implementados y utilizados en programas computacionales. Es común que los modelos resulten tener un alto nivel de complejidad, por lo que el poder de cómputo de los sistemas actuales ofrece la oportunidad de experimentar con ellos, en un tiempo de ejecución razonable.

Los modelos físicos de los instrumentos musicales están divididos conceptualmente en dos partes: mecanismo de excitación y resonador acústico [1] [2]. En los instrumentos de viento (o aliento), el mecanismo de excitación puede ser la vibración de una caña (como en el clarinete), las vibraciones de los labios (trompeta), o las oscilaciones de un chorro de aire (flauta). Generalmente, el mecanismo de excitación es modelado con una función no lineal, dado que su respuesta no depende linealmente de la variable de entrada (flujo de aire) [1].

El resonador acústico determina la altura (es decir, la entonación) del sonido que produce el instrumento. En los instrumentos de viento, el resonador se controla cerrando y abriendo distintos orificios, los cuales actúan alargando o reduciendo la longitud efectiva del tubo resonador. Esta parte puede ser modelada con una función de reflexión dependiente del tiempo [2] [3].

## ***1.2 Objetivo del proyecto de tesis***

El objetivo del proyecto actual consiste en desarrollar un programa interactivo que facilite el diseño de instrumentos musicales de viento.

Por medio de una interfaz, el usuario determina los parámetros geométricos y físicos del instrumento. Luego el programa, con los datos de entrada y con modelos físicos de instrumentos de viento, obtiene los datos acústicos más importantes (impedancia, función de reflexión, etc.) y, además, sintetiza sonidos muy similares a los que el instrumento real produciría, de ser construido.

## ***1.3 Estructura del documento de tesis***

La tesis está dividida en cuatro capítulos y una introducción. En la presente introducción se describe el tema y el contenido de la tesis.

En el primer capítulo se analiza el comportamiento sonoro de los instrumentos de viento, se discute el marco teórico de los modelos que fueron utilizados en el programa desarrollado, y se da una introducción a las señales discretas de audio digital.

En el segundo capítulo se revisa el diseño estructural del programa. Se analizan los algoritmos y las estructuras de datos implementados y se describe la interfaz visual desarrollada para la aplicación.

En el tercer capítulo se detallan los resultados del funcionamiento del programa, en pruebas comparativas con un instrumento de viento real.

En el último capítulo se resumen las principales conclusiones del proyecto, así como las recomendaciones para trabajos futuros en esta rama.

# Capítulo 1: Antecedentes

## 1.1 Simulación de instrumentos de viento

Los instrumentos de viento son todos aquellos cuya fuente de energía para generar sonido es una entrada de flujo de aire. Existen varios tipos de instrumentos de viento. Si se les clasifica por su mecanismo de producción de sonido, se tienen tres tipos fundamentales [4]:

- Por vibración de una caña flexible (clarinete, oboe, etc.).
- Por vibración de los labios (trompeta, tuba, etc.).
- Por oscilación de un chorro de aire a través de una abertura (flauta, silbato, tubos de órgano, etc.).

En la actualidad ya existen varias propuestas para la síntesis físico-acústica del sonido de diversos instrumentos de viento, tanto en el modelado matemático, como en la implementación computacional [5].

En el presente trabajo, es objeto de estudio el tercer tipo de instrumento, por ser el más fácil de evaluar para las pruebas finales del proyecto (véase el capítulo 3).

En 1982, M. E. McIntyre, et al. [1] proponen una estructura algorítmica para la simulación de distintos instrumentos de tipo tonal, así como algunos modelos simplificados de simulación. Cabe señalar que la estructura propuesta por ellos es la que se adapta al presente trabajo.

## 1.2 Acústica de tubos

### 1.2.1 Ondas acústicas planas

El sonido se transmite en forma de ondas acústicas, las cuales son oscilaciones mecánicas que viajan a través de un medio material, con masa y elasticidad distribuidas. En el aire, las ondas acústicas son longitudinales. La descripción de la forma y propagación de las ondas de sonido puede ser muy compleja, de acuerdo a los requerimientos de problema en análisis.

Una simplificación común consiste en considerar la radiación de sonido como ondas planas. Una onda plana es aquella en la que todas las variables acústicas son función de una sola variable espacial [6]. Cuando las ondas sonoras son generadas por una fuente pequeña (en

comparación con la longitud de del sonido que produce), puede considerarse que tales ondas se extienden en todas direcciones, de manera esférica. Si se toma una pequeña sección de onda retirada de la fuente, a una distancia mucho mayor que la longitud de onda, los frentes de onda pueden ser tratados como ondas planas en la dirección de propagación (véase la figura 1.1).

Supóngase que se tienen dos frentes de onda, como se puede ver en la figura 1.2. Ambos frentes con un área superficial  $S$ , y una separación  $dx$ . Si llamamos  $\xi$  al desplazamiento del aire entre estos dos planos, de tal manera que los puntos  $ABCD$  se muevan a las posiciones  $A'B'C'D'$  (como se indica en la figura 1.2), entonces el volumen de este elemento está dado por:

$$V + dV = Sdx \left( 1 + \frac{\partial \xi}{\partial x} \right) \quad (1.1)$$

Si  $P_a$  es la presión total del aire,  $p$  la pequeña parte que corresponde a la presión acústica, y se utiliza la definición del módulo de elasticidad volumétrica  $K$ , puede obtenerse la siguiente relación [4]:

$$p = dP_a = -K \frac{dV}{V}$$

$$p = -K \frac{\partial \xi}{\partial x} \quad (1.2)$$

Si se aplica la segunda ley de Newton al movimiento del elemento ABCD, señalando la fuerza del gradiente de presión en la dirección opuesta a la aceleración, se obtiene:

$$-S \left( \frac{\partial p}{\partial x} dx \right) = \rho S dx \frac{\partial^2 p}{\partial t^2}$$

$$-\frac{\partial p}{\partial x} = \rho \frac{\partial^2 \xi}{\partial t^2} \quad (1.3)$$

A partir de (1.1), (1.2) y (1.3) se puede obtener la ecuación lineal de onda acústica:

$$\frac{\partial^2 p}{\partial t^2} = c^2 \frac{\partial^2 p}{\partial x^2} \quad (1.4)$$

En donde:

$$c^2 = K / \rho \quad (1.5)$$

La cantidad  $c$  es conocida como la velocidad propagación de las ondas sonoras [4].

De este modo, se relaciona la conservación de masa y momento lineal, junto con una ecuación de estado, para obtener la ecuación lineal de onda [4].

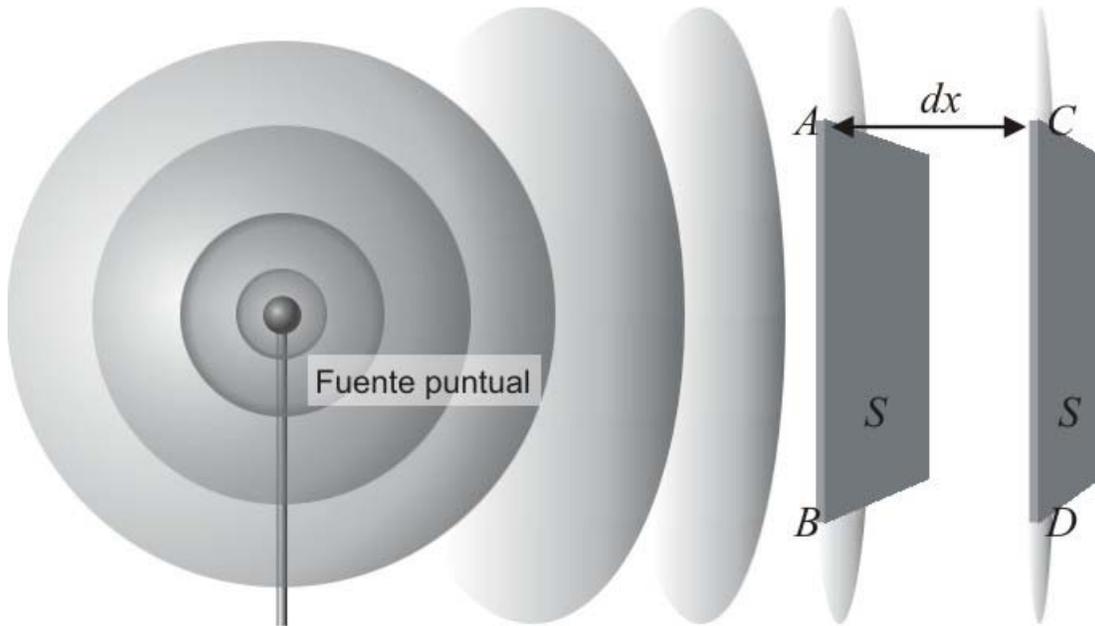


Figura 1.1: Frentes esféricas de onda acústica. Nótese que a posiciones lejanas de la fuente, se tienen frentes de onda casi planos.

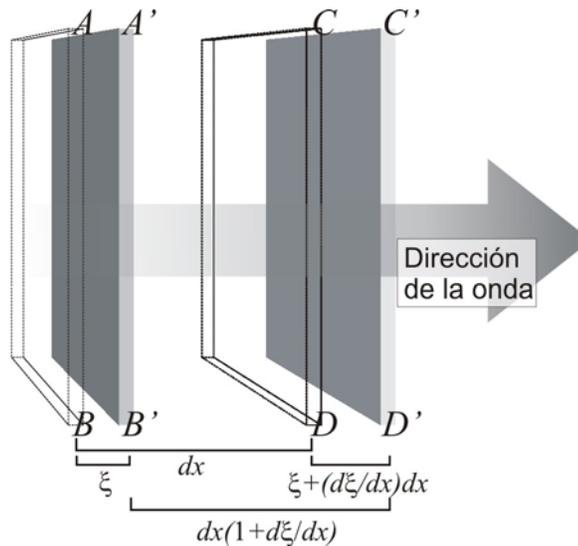


Figura 1.2: Frentes de onda acústica plana.

Es fácil verificar que las soluciones de esta ecuación diferencial tienen la forma:

$$p(x, t) = f_1(x - ct) + f_2(x + ct) \quad (1.6)$$

En donde  $f_1$  y  $f_2$  son funciones espaciales arbitrarias  $f(x-x_0)$  que se propagan en direcciones  $+x$  y  $-x$  (respectivamente) con velocidad  $c$ .

La solución de la ecuación (1.4) en el dominio de la frecuencia tiene la forma:

$$p = Ae^{-jkx}e^{j\omega t} + Be^{jkx}e^{j\omega t} \quad (1.7)$$

En donde  $k = \omega/c$  es el número de onda y los términos  $A$  y  $B$  representan ondas viajando hacia la derecha y hacia la izquierda del eje  $x$  respectivamente [4].

### 1.2.2 Impedancia acústica específica

La impedancia es normalmente definida como la oposición o resistencia al movimiento de un sistema hacia ciertos componentes de frecuencia de una señal de fuerza de entrada. Los componentes menos atenuados se denominan resonancias. En el caso del sonido que se transmite a través de un sistema, se habla de la impedancia acústica característica, la cual es la relación entre la presión (esfuerzo volumétrico) y la velocidad o flujo acústico, como se observa en (1.10).

Si se considera una onda de frecuencia angular  $\omega$  viajando en la dirección  $+x$ , entonces  $B = 0$  y  $A = 1$  en la ecuación (1.6) y se puede escribir:

$$p = e^{-jkx}e^{j\omega t} \rightarrow \cos(-kt + \omega t) \quad (1.8)$$

Donde  $\cos(-kt + \omega t)$  es la parte real de  $P$ . Se pueden relacionar  $P$  y  $\xi$  a través de la ecuación (1.3), con lo cual se obtiene:

$$jkp = j\rho\omega \frac{\partial \xi}{\partial t} \quad (1.9)$$

Siendo  $u = \partial \xi / \partial t$  la velocidad del flujo acústico, y recordando que  $k = \omega/c$ , resulta:

$$p = \rho c u \quad (1.10)$$

Con este resultado es evidente que, para ondas planas, la presión acústica y la velocidad de flujo acústico (que corresponde a la velocidad de partícula), están en fase en la dirección de propagación. Es útil definir una cantidad  $z$ , llamada impedancia de onda o impedancia acústica específica:

$$z = \frac{p}{u} = \rho c \quad (1.11)$$

Donde  $p$  es la presión acústica,  $u$  es la velocidad del fluido acústico,  $\rho$  es la densidad del medio y  $c$  es la velocidad del sonido. Esta relación se cumple únicamente para una onda plana que viaja en dirección positiva (gráficamente, hacia la derecha) [4].

### 1.2.3 Coeficientes de reflexión y transmisión

La impedancia de onda puede utilizarse para determinar la reflexión o transmisión acústicas en la interfaz de dos medios. Considérese el comportamiento de una onda de presión sonora  $Ae^{-jkx}$ , moviéndose desde un medio de impedancia  $z_1$  a otro de impedancia  $z_2$ . Se espera obtener una onda reflejada  $Be^{jkx}$  (que se mueve en dirección contraria) y una transmitida  $Ce^{-jkx}$ . Por continuidad, la presión acústica de cualquier parte de la interfaz debe ser la misma. Esto es, tomando la interfaz en  $x = 0$ :

$$A + B = C \quad (1.12)$$

De igual forma, las velocidades de desplazamiento también deben de ser las mismas en cualquier punto de la interfaz. Luego:

$$\frac{A - B}{z_1} = \frac{C}{z_2} \quad (1.13)$$

Con las ecuaciones (1.12) y (1.13) se puede calcular la proporción de la amplitud de la onda reflejada ( $C_r$ , coeficiente de reflexión) y la proporción de la amplitud transmitida ( $C_t$ , coeficiente de transmisión) [4]:

$$C_r = \frac{B}{A} = \frac{z_2 - z_1}{z_2 + z_1} \quad (1.14)$$

$$C_t = \frac{C}{A} = \frac{2z_2}{z_2 + z_1} \quad (1.15)$$

Estos coeficientes se refieren a amplitudes de presión [4].

### 1.2.4 Impedancia y reflexión a la entrada de un tubo

Los instrumentos musicales de viento frecuentemente están compuestos de tubos de longitud finita. Estos tubos presentan una impedancia acústica que provoca la resonancia de ciertos tonos. En el extremo final de un tubo, ya sea abierto o cerrado, ocurre el fenómeno de reflexión acústica.

Supóngase un tubo hueco que se extiende desde  $x=0$  hasta  $x=L$ , y que en  $x=L$  presenta la impedancia  $Z_L$ . La presión acústica dentro del tubo es una superposición de dos ondas que se mueven, una hacia la derecha y la otra hacia la izquierda, con amplitudes  $A$  y  $B$ , respectivamente. Las amplitudes son cantidades complejas, pues contienen un factor de fase. Luego, en un punto  $x$ , se tiene que la presión acústica es:

$$p(x,t) = p(x,\omega)e^{j\omega t} = (Ae^{-jkx} + Be^{jkx})e^{j\omega t} \quad (1.16)$$

La velocidad de volumen (flujo acústico) asociada a la onda de presión acústica, es también una superposición de los flujos asociados a ambas ondas. Tomando en cuenta la ecuación de movimiento  $jk\rho cU/S = -\partial p/\partial x$ , donde  $S$  es la sección transversal del tubo, se tiene que el flujo acústico es:

$$U(x,t) = U(x,\omega)e^{j\omega t} = (S/\rho c)(Ae^{-jkx} - Be^{jkx})e^{j\omega t} \quad (1.17)$$

De aquí se puede ver que en la onda que viaja en sentido positivo de  $x$ , la presión (de amplitud  $A$ ) y el flujo acústico ( $AS/\rho c$ ) tienen la misma fase; mientras que en la onda que viaja en el sentido negativo, las fases son opuestas ( $B$  y  $-BS/\rho c$ , respectivamente).

En el extremo  $x=L$ , presión y flujo se relacionan con la impedancia de terminación  $Z_L$ :

$$\frac{p(L,\omega)}{U(L,\omega)} = Z_L \quad (1.18)$$

Esta ecuación permite determinar el coeficiente de reflexión  $C_r = B/A$ . Si la impedancia característica del tubo (impedancia en  $x=0$ ), es  $Z_0 = \rho c/S$ , luego [4]:

$$C_r = \frac{B}{A} = e^{-2jkL} \frac{Z_L - Z_0}{Z_L + Z_0} \quad (1.19)$$

### 1.2.5 Impedancia y reflexión en un tubo cilíndrico

En un tubo cilíndrico, la impedancia de entrada  $Z_{in}$  en  $x=0$  (tomando en cuenta la expresión (1.19)) está dada por [4]:

$$Z_{in} = Z_0 \frac{A+B}{A-B} = Z_0 \frac{Z_L \cos(kL) + jZ_0 \sin(kL)}{jZ_L \sin(kL) + Z_0 \cos(kL)} \quad (1.20)$$

Para el caso de un tubo rígido tapado en  $x=L$ , es decir  $Z_L = \infty$ , se tiene:

$$Z_{in}^{tapado} = -jZ_0 \cot(kL) \quad (1.21)$$

En el caso de un tubo idealmente abierto, con  $Z_L = 0$ , se tiene:

$$Z_{in}^{abierto} = -jZ_0 \tan(kL) \quad (1.22)$$

Este último caso es muy difícil de obtener en la realidad, debido a la componente de impedancia de radiación, que ocurre en todos los extremos abiertos de un conducto acústico [4].

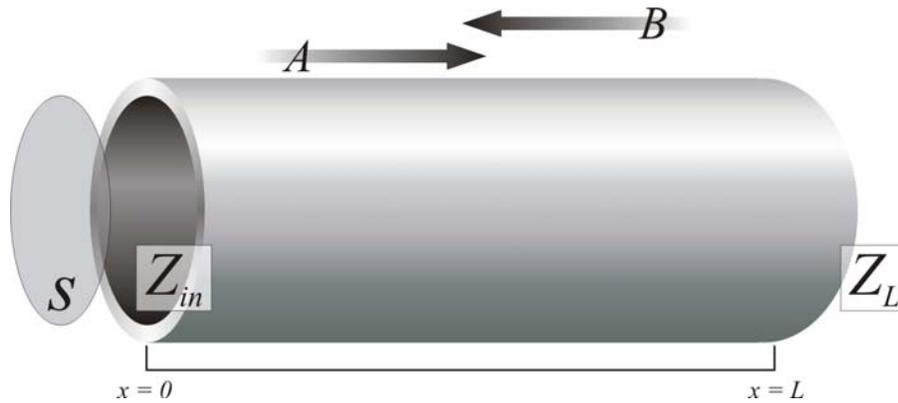


Figura 1.3 Sección tubular cilíndrica:

### 1.2.6 Impedancia en un tubo cónico

Si se considera un cono como el que se muestra en la figura 1.4, con un área inicial  $S_1$  en la posición  $x_1$  y un área final  $S_2$  en la posición  $x_2$ , cuya longitud es  $L = x_2 - x_1$ , y se hace un desarrollo como aquél para obtener la expresión (1.20), pero utilizando ondas acústicas esféricas, de tal forma que se obtiene [4]:

$$Z_{in} = \frac{\rho c}{S_1} \frac{jZ_L [\sin(kL - \theta_2) / \sin(\theta_2)] + (\rho c / S_2) \sin(kL)}{Z_L [\sin(kL + \theta_1 - \theta_2) / \sin(\theta_1) \sin(\theta_2)] - (j\rho c / S_2) [\sin(kL + \theta_1) / \sin(\theta_1)]} \quad (1.23)$$

En donde  $\theta_1 = \tan^{-1} kx_1$ ,  $\theta_2 = \tan^{-1} kx_2$ , y tanto el punto  $x_1$  como el  $x_2$  comienzan a medirse a partir del ápice del cono a lo largo del eje  $x$  (véase la figura 1.4).

Para el caso de un cono de longitud  $L$ , con un extremo idealmente abierto ( $Z_L = 0$ ), para cualquiera de los extremos de cono, se obtiene lo siguiente [4]:

$$Z_{in} = j \left( \frac{\rho c}{S_1} \right) \frac{\sin(kL) \sin(\theta_1)}{\sin(kL + \theta_1)} \quad (1.24)$$

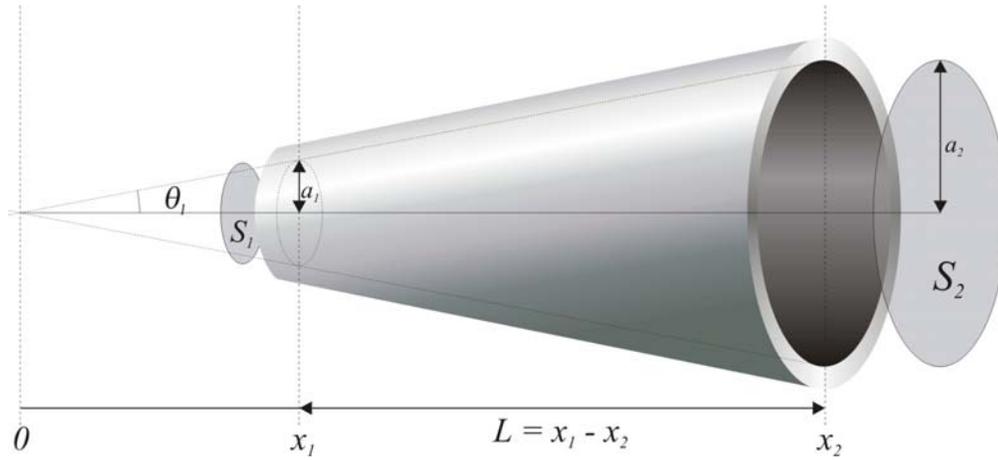


Figura 1.4: Sección tubular cónica truncada.

Lo que no implica que la impedancia de entrada sea igual en ambos extremos, ya que el signo de  $L$  y la magnitud de  $\theta_1$  serán diferentes en cada caso [4].

### 1.2.7 Función de reflexión

Sea  $p(t)$  la presión observada en la boca de un tubo cuando el flujo acústico hacia adentro es  $U(t)$ . En el dominio de la frecuencia, se puede transformar el flujo  $U(t)$  con una integral de Fourier, con lo cual se obtiene la función  $U(\omega)$ . Utilizando la definición de la impedancia de entrada  $Z(\omega) = P(\omega)/U(\omega)$ , se puede escribir [4]:

$$p(t) = \int_{-\infty}^{+\infty} Z(\omega)U(\omega)e^{j\omega t} d\omega \quad (1.25)$$

Alternativamente, se define la función de respuesta impulso o función de Green  $G(t-t')$ , la cual es la respuesta de presión al tiempo  $t$  a un impulso unitario de flujo al tiempo  $t'$ . La integral de convolución es [4]:

$$p(t) = \int_{-\infty}^t G(t-t')U(t')dt' \equiv G(t) * U(t) \quad (1.26)$$

Estos resultados son formalmente equivalentes, por lo tanto  $G(t)$  y  $Z(\omega)$  son funciones transformadas de Fourier. Supóngase que el flujo entra al instrumento desde un tubo infinito con una sección transversal  $S$ . La función de reflexión de la onda plana  $R(\omega)$ , a la entrada del tubo, puede ser expresada como:

$$R(\omega) = \frac{Z(\omega) - Z_0}{Z(\omega) + Z_0} \quad (1.27)$$

En donde  $Z_0 = \rho c / S$  es la impedancia característica a la entrada, y donde la función de reflexión en dominio del tiempo  $r(t)$  se obtiene con la transformada inversa de Fourier de  $R(\omega)$ , es decir,  $r(t) = \mathfrak{T}^{-1}\{R(\omega)\}$ .

Despejando  $Z(\omega)$  de (1.27), aplicando la transformada inversa de Fourier, y relacionando con (1.26) se obtiene:

$$\begin{aligned}
 p(t) &= Z_0 U(t) + Z_0 r(t) * U(t) + r(t) * p(t) \\
 p(t) &= Z_0 U(t) + \int_0^{\infty} r(t') [Z_0 U(t-t') + p(t-t')] dt'
 \end{aligned}
 \tag{1.28}$$

Donde el límite inferior de la integral asegura que  $r(t) = 0$  para  $t < 0$ . Esto debido a que los valores presentes al tiempo  $t$  de las variables involucradas, sólo dependen de valores pasados, por lo que el retardo no toma valores negativos. Es decir,  $t' \geq 0$  (principio de causalidad). Esta ecuación integral permite relacionar la presión y el flujo acústico de un sistema, con la función de reflexión que lo caracteriza [4].

### 1.3 Modelos físicos de los instrumentos de viento

#### 1.3.1 Excitador - resonador

Una de las primeras propuestas de modelado de este tipo de instrumentos fue realizada por McIntyre, Schumacher y Woodhouse [1]. En esta propuesta, el instrumento musical se separa en dos elementos fundamentales, uno lineal y uno no lineal (véase la figura 1.5).

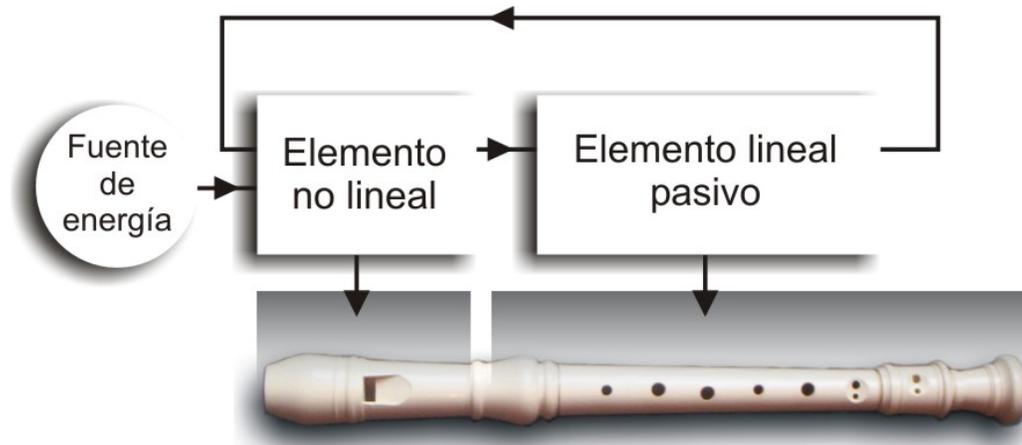


Figura 1.5: Esquema del modelo de oscilador musical, y correspondencia con las partes de una flauta dulce soprano.

El elemento que recibe la energía directamente del exterior reacciona ante la excitación de una forma marcadamente no lineal, de tal modo que las ecuaciones que lo modelan son no lineales (por ejemplo, las relaciones entre presión y flujo en la embocadura de una flauta). La dinámica de la excitación es influida por un segundo elemento (resonador), que la induce a producir un sonido tonal, haciendo oscilar el elemento resonante a ciertas frecuencias específicas. En general, el tono está relacionado linealmente con alguna característica física del elemento resonador (por ejemplo, su longitud). Al final del ciclo, parte de la energía regresa al elemento no lineal en forma de retroalimentación. Prácticamente todos los instrumentos musicales tonales de excitación constante pueden ser adaptados a este modelo.

En el caso de los instrumentos de viento, el primer elemento corresponde a la boquilla. Ahí, el movimiento del aire obedece a leyes de flujo no lineales. En las boquillas de tipo flauta, en un principio, un chorro de aire proveniente del canalillo cruza una apertura, la cual es también llamada “boca” o “ventana” (véase la figura 1.6).

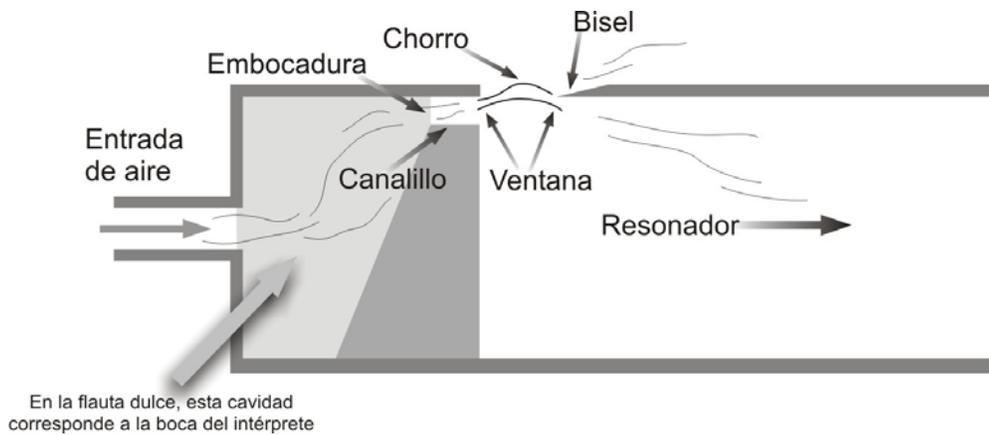
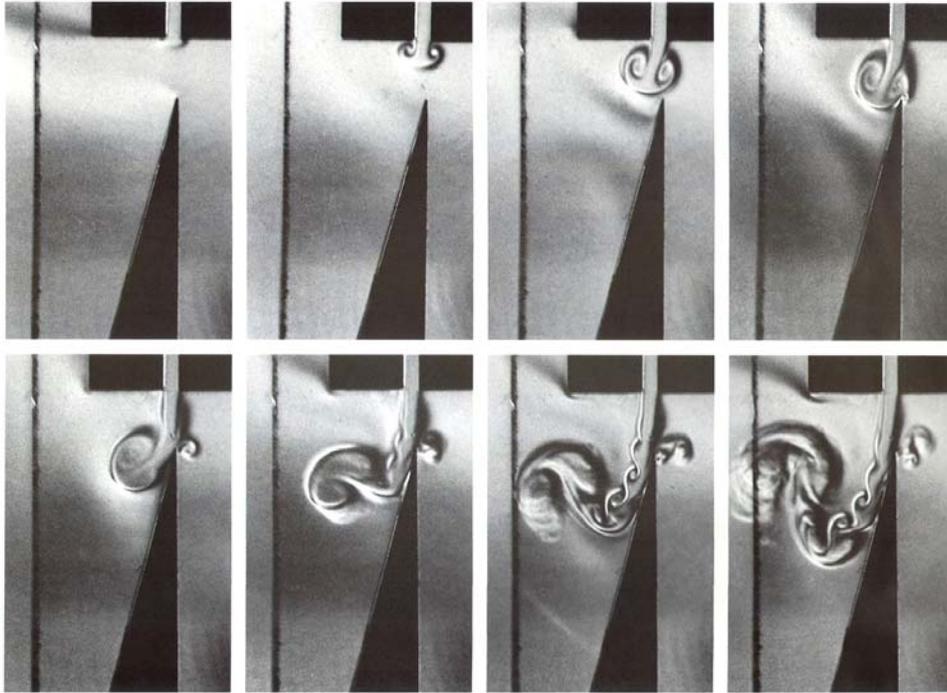


Figura 1.6: Esquema simplificado de un instrumento de viento de excitación tipo flauta.

Por la ley de Bernoulli, la presión del chorro es inferior a la presión atmosférica, pues tiene una velocidad mayor. Esto, aunado a la geometría del bisel, así como a un leve efecto de vorticidad (curvatura de los flujos), provoca una ligera desviación de la dirección del chorro, hacia fuera del tubo (véanse las cuatro primeras ilustraciones de la figura 1.7).

Una parte del chorro choca con el filo de un bisel, lo que produce una perturbación del chorro que viaja en dirección contraria y llega hasta su origen, a la salida del canalillo. Estas perturbaciones se amplifican a través del chorro, hacia el bisel, e incitan que el chorro se desvíe dentro y fuera del tubo de una forma oscilatoria. Esta excitación energética es transferida al segundo elemento, el tubo resonador.

En este elemento, se favorece la excitación en una frecuencia (tono) que está relacionada con la longitud y la geometría del tubo. La presión de aire regresa al excitador, y provoca que el chorro oscile en la frecuencia de resonancia (véanse las cuatro últimas ilustraciones de la figura 1.7). De esa forma, el resonador influye en las desviaciones del chorro de aire en la ventana, con lo cual se retroalimenta el sistema [7].



*Figura 1.7: Secuencia de fotografías del proceso inicial de oscilación de un chorro en una boquilla de tipo flauta dulce. Las imágenes fueron obtenidas del artículo “Jet formation and jet velocity fluctuations in a flue organ pipe” de Verge et al. [7], tomadas a un intervalo de 0.5 microsegundos.*

McIntyre et al. [1] propusieron un modelo físico simple en el dominio del tiempo, que caracteriza la boquilla con una relación funcional no lineal entre dos variables  $f = f(p)$ , que son el flujo acústico de aire dentro de la boquilla  $f(t)$  y la presión acústica de aire  $p(t)$ . Por otra parte, el tubo del instrumento se caracteriza con una función de reflexión  $r(t)$ , que describe la relación entre la onda de presión sonora que viaja hacia el tubo  $p_o(t)$ , y la que llega reflejada desde el tubo hacia la embocadura  $p_i(t)$ . Estas variables se relacionan de la siguiente manera:

$$p_i(t) = r(t) * p_o(t) \quad (1.29)$$

$$p(t) = p_o(t) + p_i(t) \quad (1.30)$$

$$Z_0 f(t) = p_i(t) - p_o(t) \quad (1.31)$$

Donde el operador \* indica convolución, y  $Z_0$  es la impedancia acústica característica en la embocadura. Este modelo permite simular la evolución temporal de la presión sonora en la embocadura en términos del elemento excitador (con la relación entre presión sonora y flujo  $f = f(p)$ ); y del elemento resonador, (con la función de reflexión del tubo  $r(t)$  [3]).

### 1.3.2 La boquilla: Modelo de McIntyre et al.

La boquilla es el elemento excitador no lineal. Para instrumentos de viento de excitación tipo flauta dulce, se consideraron dos modelos físicos: el propuesto por McIntyre et al. [1] y el propuesto por Verge et al. [9].

En el primer modelo, se tiene en cuenta que la excitación generada por oscilaciones del chorro de aire, al chocar con un bisel, provoca perturbaciones sinuosas (véase la figura 1.7), que son las que dan origen a la oscilación inicial.

Puesto que estas perturbaciones viajan desde el canalillo hasta el bisel a una velocidad finita, existe un “tiempo de retraso” en el proceso de retroalimentación del sistema, de tal forma que [1]:

$$f(t) = F\{p(t - \tau)\} \quad (1.32)$$

Donde  $\tau$  es el retraso temporal, que puede calcularse con la velocidad del chorro  $U_j$  y la distancia del canalillo al bisel  $W$  (la anchura de la ventana) de tal forma que:

$$\tau = U_j / W \quad (1.33)$$

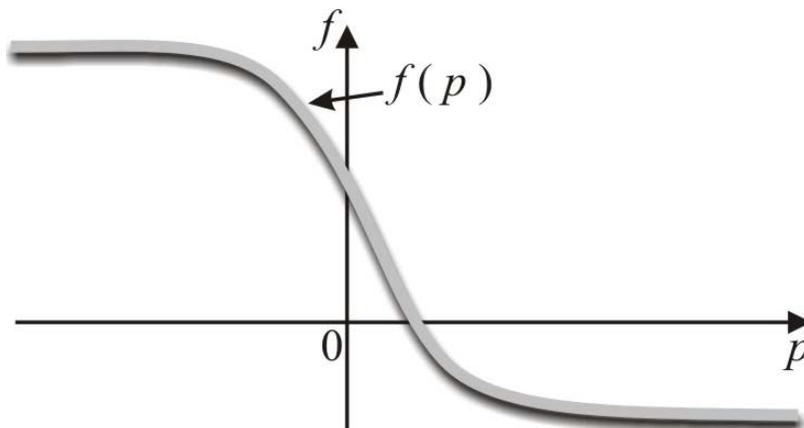


Figura 1.8: Función de saturación que representa la dependencia modelada entre el flujo volumétrico y la presión acústica.

Suponiendo que la presión manométrica a la salida del canalillo sea cero, la velocidad del chorro  $U_j$  puede calcularse a partir de la presión a la entrada del canalillo  $P$  y la densidad media del aire  $\rho$ , con la siguiente expresión [9]:

$$U_j = \sqrt{2P/\rho} \quad (1.34)$$

La relación  $f = f(p)$  propuesta por el modelo, es una función con la forma de una tangente hiperbólica. Haciendo una adaptación a lo propuesto por Fletcher [4], se propone la siguiente relación:

$$f(p) = WbU_j \left[ 1 + \tanh\left(l \frac{p(t-\tau)}{b}\right) \right] \quad (1.35)$$

Donde  $W$  es la distancia entre el canalillo y el borde del bisel,  $U_j$  es la velocidad central del chorro a la salida del canalillo,  $l$  es una constante de proporcionalidad y  $b$  es la anchura del chorro (que puede aproximarse al ancho del canalillo, pues por la corta distancia, el cambio en la anchura del chorro desde el confinamiento del canalillo al filo del bisel es despreciable, en términos de la magnitud de  $b$ ; véase la figura 1.8).

Es importante notar que cuando  $b$  tiende a cero, la función  $f(p)$  tiende a convertirse en una función escalón. Este límite matemático no es físicamente realista, pues la anchura del chorro no puede ser nula. El caso  $b=0$  correspondería a un flujo alternado del chorro (reducido a una lámina delgada), con transiciones abruptas entre ambos lados del bisel.

### 1.3.3 La boquilla: Modelo de Verge et al.

La estructura principal de este modelo se basa en la representación unidimensional del instrumento, la cual puede verse en la figura 1.9. La porción contraída del tubo, definida por la longitud  $\delta_m$ , representa la ventana del instrumento. A su vez,  $Q_j$  representa el flujo volumétrico del chorro, el cual es dividido por el bisel en  $Q_{in}$  y  $Q_{out}$  (los cuales son positivos cuando se dirigen hacia dentro del tubo). El salto de presión  $\Delta p$  corresponde principalmente a la división del chorro, pero puede incluir efectos secundarios de turbulencia y pérdidas debidas a vorticidades [10].

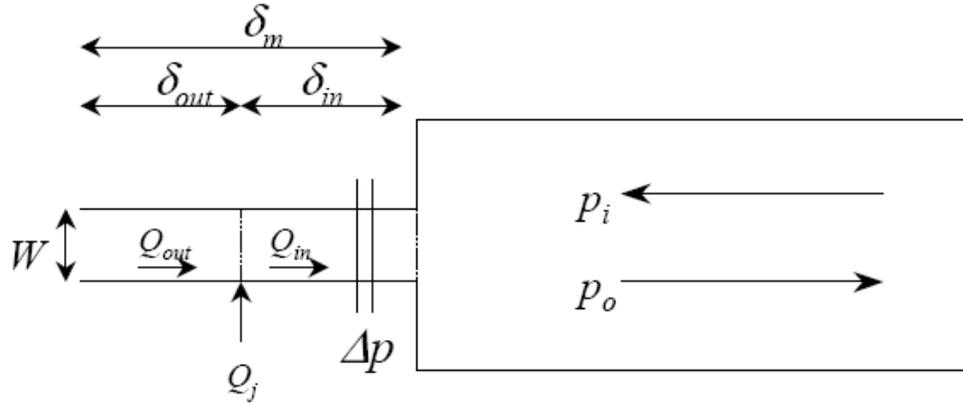


Figura 1.9: Esquema unidimensional de un instrumento tipo flauta dulce utilizando por Verge et al [9].

La ecuación principal del modelo de simulación está dada por [9]:

$$\frac{r_m^2 S_p}{4c_0 S_m} \frac{\partial^2 (p_o - p_i)}{\partial t^2} - \frac{\delta_m S_p}{c_0 S_m} \frac{\partial (p_o - p_i)}{\partial t} - (p_o - p_i) = \frac{\rho_0 r_m^2}{4c_0 S_m} \frac{\partial^2 Q_j}{\partial t^2} - \frac{\rho_0 \delta_{out}}{S_m} \frac{\partial Q_j}{\partial t} - \Delta p - 2p_i \quad (1.36)$$

Donde  $S_m$  y  $S_p$  definen las áreas transversales de la ventana y el tubo (respectivamente),  $r_m$  es el radio efectivo de la ventana y  $c_0$  es la velocidad del sonido.

Los términos del lado izquierdo de la expresión (1.36) permiten interpretarla como una ecuación diferencial de segundo orden con la variable incógnita  $(p_o - p_i)$ , donde los términos del lado derecho pueden ser considerados factores de forzamiento. El lado izquierdo de la ecuación (1.36) implica modos de evolución dinámica exponencialmente divergentes (amortiguamiento negativo en el segundo término del lado izquierdo), lo cual puede dar lugar a dificultades en la solución numérica de la ecuación.

La variable  $\Delta p$  es bastante compleja. Se compone de la suma de varios elementos: el salto de presión producido por la división del chorro  $\Delta p_{jd}$ , el salto de presión debido a la turbulencia, el salto que arroja la vorticidad del flujo, e incluso un pulso transitorio inicial, que corresponde al impacto inicial del chorro en contra del bisel. De todos los elementos,  $\Delta p_{jd}$  es el más relevante en el modelo, pues es el que provee casi toda la energía hacia el tubo, que provoca la oscilación. El flujo de presión debido a la división del chorro se modela como dos fuentes de presión complementarias, una en cada lado del bisel. De esta forma, el salto de presión se deriva de la siguiente forma:

$$\Delta p_{jd} = -\frac{\rho_0 \delta_d}{S_m} \frac{dQ_i}{dt} \quad (1.37)$$

Donde  $\delta_d$  se define como la distancia “acústica” entre las dos fuentes virtuales de presión  $Q_1$  y  $Q_2$ , alrededor del bisel [11] [7]. Suponiendo que el chorro tiene un perfil de velocidades de Bickley [4],  $Q_1$  puede definirse como:

$$Q_1 = bHU_0 \left[ 1 + \tanh\left(\frac{\eta - y_0}{b}\right) \right] \quad (1.38)$$

Donde  $b$  es la mitad de la anchura del chorro,  $H$  es la anchura del tubo resonador,  $U_0$  es la velocidad central del chorro y  $y_0$  es el desplazamiento del centro del chorro con respecto al filo del bisel. Aquí puede verse que se incluye la función de saturación, presente en el modelo anterior. Finalmente,  $\eta$  es el desplazamiento del chorro frente al bisel, y es una función tanto de la longitud del chorro  $W$ , como de la frecuencia de perturbación  $\omega$ .

La ecuación para  $\eta$  está basada en una modificación de la fórmula de desplazamiento del chorro presentada por Fletcher y Thwaites [9] y está dada por:

$$\eta(W, \omega) = \underbrace{\frac{1}{i\omega} \left( \frac{2 Q_p}{\pi S_m} - \frac{0.38 Q_1'}{S_m} \right)}_1 \times \underbrace{\left[ 1 - \cosh(\mu W) e^{\left( \frac{i\omega W}{U_{pr}} \right)} \right]}_2 \quad (1.39)$$

Donde  $\mu$  es el coeficiente de amplificación de las perturbaciones del chorro,  $U_{pr}$  es la velocidad de tales perturbaciones en el chorro y  $Q_p$  es la presión de entrada dentro del resonador. Éste último puede calcularse de la siguiente manera:

$$Q_p = \frac{S_p}{\rho_0 c_0} (p_o - p_i) \quad (1.40)$$

En donde  $S_p$  es el área transversal de la entrada al resonador,  $\rho_0$  es la densidad del aire y  $c_0$  es la velocidad del sonido en el aire.

El cómputo de  $\eta$  en el dominio del tiempo se dificulta por varios factores. Primero,  $\eta$  está definido en función de la frecuencia. Además, las variables  $U_{pr}$  y  $\mu$ , también varían con la frecuencia. Para simplificar el cómputo, esta fórmula puede implementarse haciendo una aproximación con un filtro digital de tipo Butterworth [9].

La fórmula (1.39) tiene un comportamiento no realista cuando la frecuencia  $\omega$  es igual a 0. Un filtro digital pasa banda puede aproximar la amplitud de respuesta del chorro, sin incurrir en esta singularidad matemática [9].

El factor (1) corresponde a la operación de integración y filtrado de la señal. El factor (2) incluye una operación de retardo de  $W/u$  segundos. Esta distinción entre ambos factores es aprovechada en la implementación del algoritmo computacional.

### 1.3.4 El tubo resonador

Es en el tubo resonador donde la excitación de la boquilla es inducida a una dinámica que corresponde a un tono armónico. Un conjunto de impulsos, en forma de presión de entrada  $p_i$ , se introduce al resonador, recorre el tubo, rebota en un orificio abierto y regresa como presión de salida  $p_o$ , con sentido inverso a  $p_i$ . Las transformaciones de la presión de salida con respecto a la de entrada dependen directamente de las características geométricas del tubo y sus orificios [12] [13]. Por lo tanto, la función de reflexión del tubo resonador  $r(t)$  dependerá de tales características geométricas.

Keefe [12] propone un método para obtener la función de reflexión a través de la geometría conocida del resonador, por medio de matrices de transferencia. El método consiste en separar el tubo en elementos de distintos tipos: cilíndricos, cónicos y de orificio, conectados de manera simple. Cada elemento tiene un papel dentro de la generación de la función de reflexión, y este papel es representado por una matriz.

El enfoque básico es similar a la teoría de las redes eléctricas. Sustituyendo el voltaje por la presión acústica  $p$ , la corriente eléctrica por el flujo de volumen acústico  $F$ , y tomando en cuenta que ambas cantidades oscilan en el tiempo con una frecuencia angular  $\omega$ .

Considérese un sistema acústico arbitrario de dos puertos, una entrada y una salida. Se denomina  $T$  la matriz de transferencia que relaciona el flujo volumétrico  $f_1$  y presión  $p_1$  de entrada con el flujo de volumen  $f_0$  y presión  $p_0$  de salida. Si se denominan  $A$ ,  $B$ ,  $C$  y  $D$  los coeficientes complejos de la matriz de transferencia  $T$ , se tiene que:

$$\begin{pmatrix} p_0 \\ f_0 \end{pmatrix} = \underbrace{\begin{pmatrix} A & B \\ C & D \end{pmatrix}}_T \begin{pmatrix} p_i \\ f_i \end{pmatrix} \quad (1.41)$$

Así,  $T_b(L, a_1, \theta)$  denota una matriz de transferencia para la propagación de onda dentro de un tubo cónico truncado con paredes lisas, con una longitud  $L$  y radio  $a_1$  a la entrada. El ángulo de media apertura del cono es  $\theta$ . Por lo tanto, el ángulo de apertura total es  $2\theta$  [12].

### 1.3.5 Matrices de transferencia de cilindros y conos

Considerando un cilindro simple, de longitud  $L$ , radio  $a_1$ , y para el cual  $\theta = 0$ , la matriz de transferencia  $T_b(L, a_1, 0)$  es:

$$T_b(L, a_1, 0) = \begin{pmatrix} \cosh \Gamma L & Z_c \sinh \Gamma L \\ Z_c^{-1} \sinh \Gamma L & \cosh \Gamma L \end{pmatrix} \quad (1.42)$$

Las pérdidas viscosas y térmicas en las paredes del tubo son tomadas en cuenta en términos de la constante de propagación compleja  $\Gamma = jk$  (donde  $k$  es el número de onda) y la impedancia característica  $Z_c$ .

El tubo cónico truncado de longitud  $L$  posee una entrada de radio  $a_1$  y un radio  $a_2$  en su extremo opuesto. Expresando  $a_2$  en términos de la longitud  $x_1$ , se obtiene la relación con  $\theta$ , la cual es:

$$\tan \theta = a_1 / x_1 = a_2 / (L + x_1) \quad (1.43)$$

Donde  $x_1$  es la distancia a lo largo del eje  $x$  del tubo cónico desde el ápice del cono completo hipotético a la entrada del cono truncado (véase la figura 1.4). Los coeficientes de la matriz de transferencia  $T_b(L, a_1, \theta)$  para el tubo cónico se expresan utilizando (1.33), con lo que se obtienen las siguientes expresiones:

$$A = \left( \frac{x_1 + L}{x_1} \right) \cosh \Gamma L - \frac{\sinh \Gamma L}{[\Gamma(x_1 + L)]} \quad (1.44)$$

$$B = \left( \frac{x_1}{x_1 + L} \right) Z_c \sinh \Gamma L \quad (1.45)$$

$$C = Z_c^{-1} \left[ \frac{x_1 + L}{x_1} - \left( \frac{1}{\Gamma x_1} \right)^2 \right] \sinh \Gamma L + \frac{\Gamma L}{(\Gamma x_1)^2} \cosh \Gamma L \quad (1.46)$$

$$D = \left( \frac{x_1}{x_1 + L} \right) \left[ \cosh \Gamma L - \frac{1}{(\Gamma x_1)} \sinh \Gamma L \right] \quad (1.47)$$

La exactitud de este método es mejor para tubos largos, si éstos se separan y analizan como varios segmentos más cortos [12].

### 1.3.6 Matrices de transferencia de agujeros de digitación

El comportamiento acústico de los instrumentos de viento puede representarse como una línea de transmisión en una dimensión, en la cual los agujeros de digitación abiertos o cerrados se representan como elementos concentrados.

Considérese un agujero de digitación cilíndrico de radio  $b$ , localizado en un tubo cilíndrico o cónico de radio  $a$ , como se muestra en la figura 1.10. Para las dimensiones de los tubos cónicos comunes en los instrumentos musicales, es suficientemente preciso ignorar la conicidad en una distancia del orden del diámetro del agujero de digitación. Dado que la superficie de intersección entre el agujero y el tubo es de la forma de una “silla de montar”, la altura  $t$  del agujero de digitación es definida como la longitud cuyo producto con área transversal del agujero de digitación es exactamente igual al volumen geométrico  $V_h$  del agujero de digitación, es decir:

$$t = V_h / \pi b^2 \quad (1.48)$$

Éste parámetro es difícil de identificar gráficamente, por lo que no fue incluido en la figura 1.10. Un parámetro más fácil de medir es el grosor de la pared  $t_w$ , que es la diferencia entre el radio exterior y el interior del tubo (véase la figura 1.10). La altura del agujero de digitación definida en (1.48) se relaciona con  $t_w$  de la siguiente forma [12]:

$$t = t_w + (1/8)b(b/a) \left[ 1 + 0.172(b/a)^2 \right] \quad (1.49)$$

Los agujeros abiertos o cerrados se representan en el modelo como elementos concentrados de impedancias en paralelo  $Z_s$ , y de impedancias en serie  $Z_a$ . Todas las impedancias acústicas son definidas como razones entre la presión acústica y la velocidad de volumen a lo largo del eje  $x$  del tubo. La dependencia armónica del tiempo asumida es  $e^{j\omega t}$ . Entonces, el conjunto completo de impedancias para los agujeros de digitación es:

$$Z_s^{(OP)} = R_0 (a/b)^2 (jkt_e + \xi_e) \quad (1.50)$$

$$Z_s^{(CL)} = -jR_0 (a/b)^2 \cot kt_e \quad (1.51)$$

$$Z_a^{(OP)} = -jR_0(a/b)^2 kt_a^{(o)} \quad (1.52)$$

$$Z_a^{(CL)} = -jR_0(a/b)^2 kt_a^{(c)} \quad (1.53)$$

Donde  $c$  es la velocidad del sonido,  $k = \omega/c$  es el número de onda, la impedancia de onda del tubo en el agujero de digitación es  $R_0 = \rho c / \pi a^2$ ,  $\rho$  es la densidad del aire y los superíndices  $OP$  y  $CL$  denotan un agujero abierto o cerrado, respectivamente.

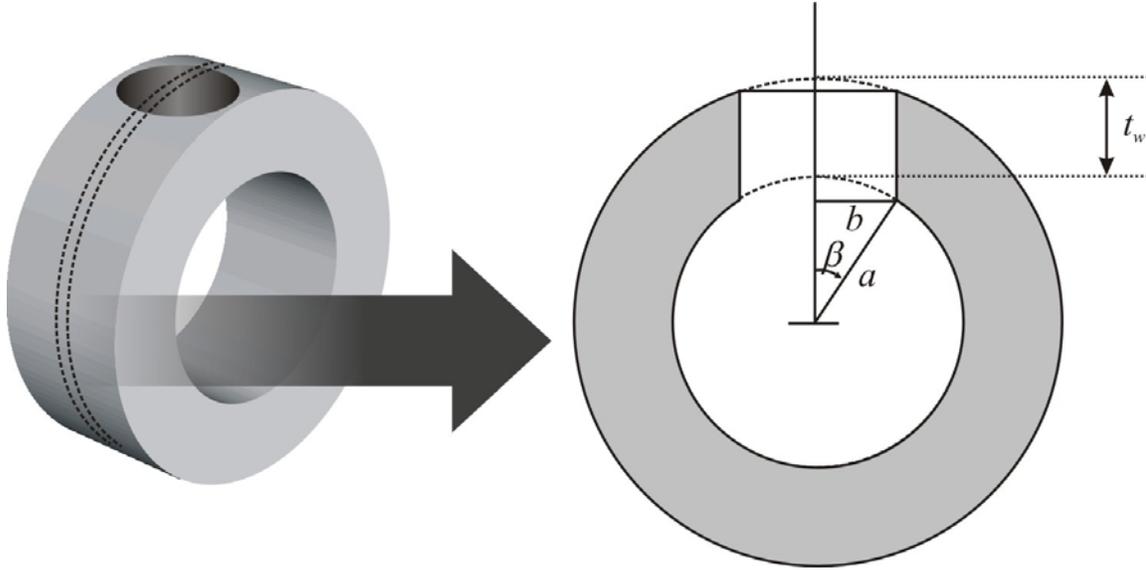


Figura 1.10: Dimensiones geométricas principales de un agujero de digitación.

La longitud efectiva del agujero de digitación  $t_e$ , las longitudes equivalentes  $t_a^{(OP)}$  y  $t_a^{(CL)}$ , y la resistencia específica  $\xi_e$  pueden definirse de la siguiente forma [12], para un agujero sin llave de digitación:

$$t_e = \frac{(1/k)\tan kt + b[1.40 - 0.58(b/a)^2]}{1 - 0.61kb \tan kt} \quad (1.54)$$

$$\xi_e = 0.25(kb)^2 + \alpha t + (1/4)kd_v \ln(2b/r_c) \quad (1.55)$$

$$t_a^{(OP)} = \frac{0.47b(b/a)^4}{\tanh(1.84t/b) + 0.62(b/a)^2 + 0.64(b/a)} \quad (1.56)$$

$$t_a^{(CL)} = \frac{0.47b(b/a)^4}{\coth(1.84t/b) + 0.62(b/a)^2 + 0.64(b/a)} \quad (1.57)$$

En donde el radio de curvatura de las terminaciones del agujero de digitación interno y externo es  $r_c$ , donde el grosor de la capa viscosa  $d_v = \sqrt{2\eta/(\rho\omega)}$  está dado en términos de viscosidad dinámica del aire  $\eta$  y donde la parte real de la propagación del número de onda dentro del agujero de digitación es  $\alpha$ . Tomando en cuenta la razón de calores específicos  $\gamma$ , la conductividad térmica del aire  $\kappa$  y el calor específico a presión constante  $C_p$ , se puede aproximar que:

$$\alpha = 1/(2bc) \left[ \sqrt{2\eta\omega/\rho} + (\gamma - 1) \sqrt{2\kappa\omega/\rho C_p} \right] \quad (1.58)$$

Finalmente, las matrices de transferencia  $T_h^{(CL)}$  y  $T_h^{(OP)}$  para un agujero cerrado o abierto, son, respectivamente [12]:

$$T_h^{(CL)} = \begin{pmatrix} 1 & Z_a^{(c)} \\ 1/Z_s^{(c)} & 1 \end{pmatrix} \quad (1.59)$$

$$T_h^{(OP)} = \begin{pmatrix} 1 & Z_a^{(o)} \\ 1/Z_s^{(o)} & 1 \end{pmatrix} \quad (1.60)$$

Para simplificar la interfaz del programa de cómputo que se describe en el siguiente capítulo, en lo sucesivo se supondrá que los agujeros de digitación son totalmente cilíndricos. Ésta suposición no abarca varios tipos de flautas dulces de diseño histórico, que poseen agujeros de digitación cónicos, abiertos hacia dentro del resonador [14]. Sin embargo, la formulación también se puede aplicar a estos casos, tomando en cuenta el volumen equivalente  $V_h$  del agujero de digitación cónico.

### 1.3.7 Función de reflexión del resonador completo

Puesto que el objetivo del modelo es obtener la función de reflexión  $r(t)$  del tubo resonador de un instrumento de viento con secciones cónicas o cilíndricas y con agujeros de digitación abiertos o cerrados, es conveniente utilizar el método de matrices de transferencia en el dominio de la frecuencia; y después aplicar la transformada inversa de Fourier para obtener la función en el dominio del tiempo.

El método de matrices de transferencia divide el tubo en secciones cilíndricas o cónicas y en secciones de longitud nula que representan agujeros de digitación cerrados o abiertos. En cada sección se produce un cambio de presión y flujo de la siguiente manera:

$$\begin{pmatrix} p_0(\omega) \\ f_0(\omega) \end{pmatrix} = \begin{pmatrix} A(\omega) & B(\omega) \\ C(\omega) & D(\omega) \end{pmatrix} \begin{pmatrix} p_1(\omega) \\ u_1(\omega) \end{pmatrix} \quad (1.61)$$

En donde los elementos de la matriz de transferencia tienen expresiones particulares para cada tipo de sección, como se ha explicado en las secciones anteriores.

El modelo completo de tubo se obtiene encadenando productos de matrices de transferencia, de tal forma que la presión y el flujo acústico a través del elemento  $n$  podrán calcularse con la función de transferencia del elemento de la siguiente forma:

$$\begin{pmatrix} p_{n+1}(\omega) \\ f_{n+1}(\omega) \end{pmatrix} = \begin{pmatrix} A_n(\omega) & B_n(\omega) \\ C_n(\omega) & D_n(\omega) \end{pmatrix} \begin{pmatrix} p_n(\omega) \\ u_n(\omega) \end{pmatrix} \quad (1.62)$$

Si la presión se supone igual a la impedancia de radiación en el extremo de un tubo cilíndrico hacia campo libre  $Z_r(\omega) = Z_0((kr)^2/4 + j0.61kr)$  (donde  $r$  es el radio de la sección transversal del tubo) [6], y el flujo se supone unitario, entonces la presión y el flujo en la embocadura se calculan de la siguiente manera:

$$\begin{pmatrix} p(\omega) \\ f(\omega) \end{pmatrix} = \begin{pmatrix} A(\omega) & B(\omega) \\ C(\omega) & D(\omega) \end{pmatrix} \begin{pmatrix} Z_r(\omega) \\ 1 \end{pmatrix} \quad (1.63)$$

Donde la matriz de transferencia es ahora el producto de las matrices de todas las secciones, como se muestra en la figura 1.11.

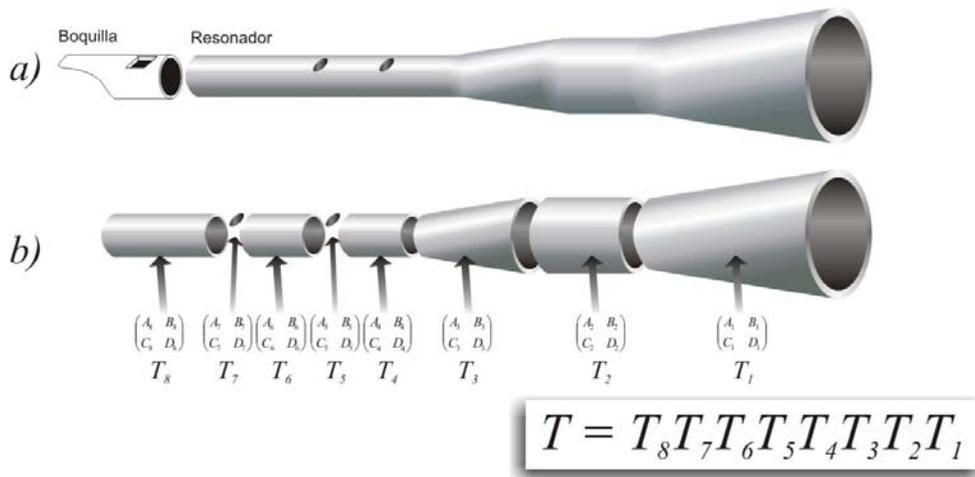


Figura 1.11: Ejemplo esquemático del método de matrices de transferencia, donde a) es el instrumento inicial y b) es el resonador dividido en elementos, cada uno con su matriz de transferencia correspondiente. La matriz  $T$  representa la transferencia conjunta de los elementos del tubo resonador.

La impedancia acústica específica del tubo completo se obtiene con la siguiente expresión:

$$Z(\omega) = \frac{p(\omega)}{f(\omega)} = \frac{A(\omega)Z_r(\omega) + B(\omega)}{C(\omega)Z_r(\omega) + D(\omega)} \quad (1.64)$$

Y el coeficiente de reflexión, en el dominio de la frecuencia (1.27), resulta:

$$R(\omega) = \frac{Z(\omega) - Z_0}{Z(\omega) + Z_0} \quad (1.65)$$

Donde  $Z_0$  es la impedancia acústica característica correspondiente a la sección inicial del tubo [12].

Finalmente, la función de reflexión  $r(t)$  se obtiene mediante la transformada inversa de Fourier de  $R(\omega)$ , de tal forma que [2]:

$$r(t) = \mathfrak{F}^{-1}\{R(\omega)\} \quad (1.66)$$

### 1.3.8 Adaptación a algoritmos digitales

En los algoritmos computacionales, es preciso convertir las funciones continuas de la matemática analítica en funciones discretas. Para ello, las funciones son evaluadas a un ritmo constante, y los resultados son almacenados en un vector de datos que representa la función ya discreta. A esta operación se le conoce como muestreo, y el ritmo de evaluación es la frecuencia de muestreo  $f_s$ , como se muestra en la figura 1.9 [15].

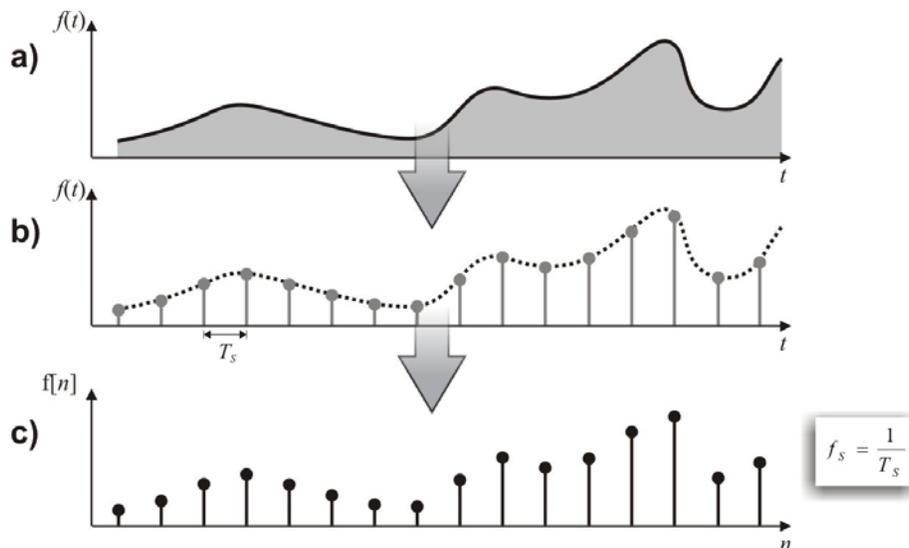


Figura 1.9: Proceso de conversión de una función continua, a una función discreta. La función continua (a) es muestreada a una frecuencia constante  $f_s$  (b) y se obtiene un vector de datos (c).

Para una función dependiente del tiempo, como es el caso de la función de reflexión  $r(t)$ , se tiene que para una muestra  $n$ :

$$n = 0, 1, 2, \dots, N-1 \quad t_n = n f_s \quad (1.67)$$

Donde  $N$  es el número total de muestras. Para una función en el dominio de la frecuencia, como el coeficiente de reflexión  $R(\omega)$ , se tiene que, para una muestra  $k$ :

$$k = 0, 1, 2, \dots, N-1 \quad \omega_k = \frac{2\pi k}{N} \quad (1.68)$$

Donde  $N$  es el número total de muestras. Por lo tanto, se definen las funciones discretas  $r[n] = \{r_0, \dots, r_n, \dots, r_{N-1}\}$  y  $R[k] = \{R_0, \dots, R_k, \dots, R_{N-1}\}$  de la siguiente forma:

$$r_n = r(t_n) \quad (1.69)$$

$$R_k = R(\omega_k) \quad (1.70)$$

De tal forma, la función de reflexión discreta  $r[n]$  y el coeficiente de reflexión discreto  $R[k]$ , en analogía con la expresión (1.60), están relacionados de la siguiente forma:

$$r[n] = \text{ift}[R[k]] \quad (1.71)$$

Donde la función  $\text{ift}[\ ]$  es la transformada inversa de Fourier [16].

La función de reflexión en el dominio del tiempo  $r[n]$  es de valores reales. Por ello, en la ecuación (1.71)  $R[k]$  debe cumplir la condición equivalente en el dominio de la frecuencia, la simetría compleja conjugada, de tal forma que:

$$R[k] = R^*[N-k], \quad k < N/2 \quad (1.72)$$

Donde se supone que  $k = N$  corresponde a la frecuencia de muestreo  $\omega_s$  y  $k = N/2$  a la frecuencia de Nyquist  $\omega_s/2$ , para evitar el fenómeno de "aliasing". Esta condición debe estar dada antes de aplicar la función  $\text{ift}[\ ]$  [15] [16].

El "aliasing" es un fenómeno en el cual una frecuencia no es muestreada correctamente debido a su cercanía con la frecuencia de muestreo. Puesto que el muestreo crítico de una función sinusoidal está dado por dos muestras (una para el valor máximo y otra para el mínimo), la frecuencia de muestreo debe ser (por lo menos) del doble de la frecuencia máxima contenida en la señal. De tal forma, la frecuencia de Nyquist establece un límite para la máxima frecuencia registrable en una señal digital [16].

Una versión óptima para el cómputo de la transformada inversa de Fourier `ifft[ ]`, es la transformada inversa rápida de Fourier `ifft[ ]`. Dado el funcionamiento del algoritmo interno de la función `ifft[ ]`, es más eficiente en la práctica, si opera sobre un vector cuyo tamaño es una potencia de dos ( $2^n$ ).

## *Capítulo 2: Desarrollo de la aplicación.*

### **2.1 Estructura del programa**

#### **2.1.1 Requerimientos de la interfaz de usuario**

Puesto que el presente programa está dirigido a usuarios dedicados a la fabricación de instrumentos del ramo, la interfaz de usuario es un elemento indispensable para su buen funcionamiento. Por lo tanto, se acordó que debía de cumplir con las siguientes características:

- Permitir la edición sencilla e intuitiva de instrumentos de tipo flauta.
- Prevenir la inserción de errores o de características incompatibles con los modelos internos del instrumento
- Guiar al usuario en las opciones menos explícitas.
- Permitir al usuario el control completo de los parámetros de los modelos.
- Ser agradable visual y operativamente.

Teniendo esto en cuenta, el programa fue construido con una interfaz gráfica, diseñada para la plataforma Windows 2000/XP, que es la más difundida en la actualidad.

La programación de los módulos y la interfaz fue realizada con la paquetería Borland Builder C++ 6.0, que permite el desarrollo rápido de aplicaciones (RAD, por sus siglas en inglés) [17]. En este tipo de programación visual, orientada a ambientes gráficos, la estructura de la aplicación se basa en ventanas (interfaz gráfica) y bibliotecas (conjunto de funciones operativas).

Dada la expectativa de poder presentar internacionalmente los módulos operativos del programa, el código fue redactado con vocablos en inglés. La interfaz, a su vez, fue creada en español.

#### **2.1.2 Estructura de ventanas y bibliotecas**

Tomando en cuenta los requerimientos de la interfaz de usuario, la estructura modular del programa fue establecida de la siguiente forma:

- **Fase de diseño:** En esta fase, el usuario introduce las dimensiones y características que definen el diseño del instrumento. El programa muestra gráficos realizados en

tiempo real, en los que están disponibles imágenes del instrumento, que cambian de acuerdo a las indicaciones del usuario. Para esta fase, son necesarias las ventanas de entrada y edición de elementos de instrumentos, así como las bibliotecas de estructura de datos (Inst\_saver.cpp) y de generación de gráficos (Inst\_draw.cpp).

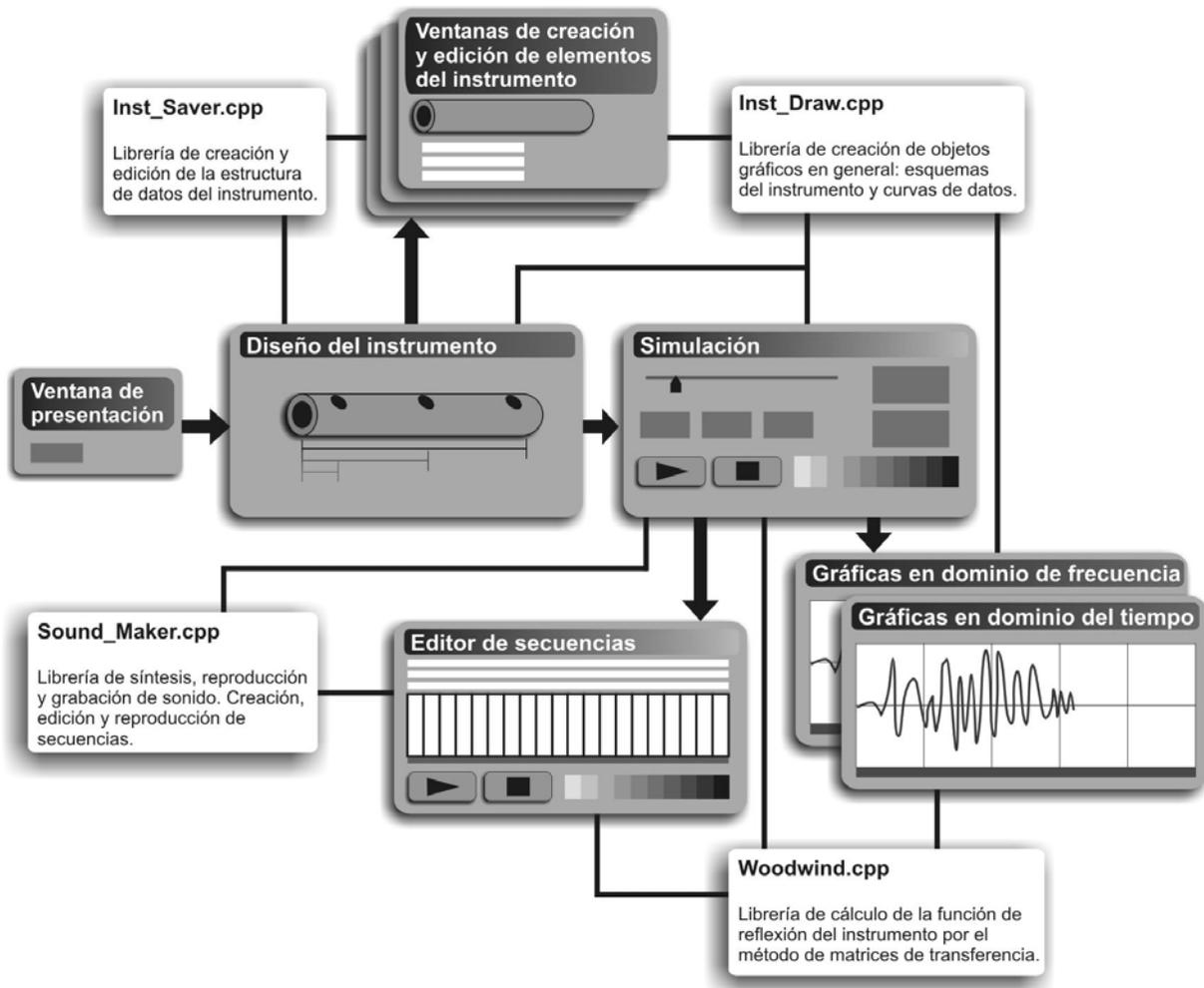


Figura 2.1: Estructura esquemática del programa, donde se especifica la relación entre las ventanas y las bibliotecas de operación.

- Fase de simulación:** Esta es la etapa en la que se utilizan los datos del diseño del instrumento, como parámetros del modelo de simulación de instrumentos de viento. El modelo da como resultado sonidos, así como la impedancia acústica, el coeficiente de reflexión y la función de reflexión. Esta fase se lleva a cabo en una ventana consecuente con la de diseño: la ventana de simulación. Se utiliza la biblioteca del modelo del resonador (Woodwind.cpp) y la biblioteca del modelo de excitación y de grabación y reproducción de sonido (Sound\_Maker.cpp). Desde esta ventana es

posible llamar las ventanas de visualización de datos (que, además, hace uso de funciones gráficas de Inst\_draw.cpp), y la ventana de ejecución de secuencias.

Esta estructura del programa se puede observar con mayor claridad en el esquema de la figura 2.1.

## 2.2 Interfaz de usuario

### 2.2.1 Ventana de bienvenida

La ventana de bienvenida se presenta al iniciar el programa, y permite elegir entre iniciar con un instrumento nuevo o abrir un instrumento existente, antes de cargar la ventana de diseño (véase la figura 2.2).



Figura 2.2: Ventana de bienvenida del programa.

### 2.2.2 Ventana de diseño

La ventana de diseño es el formulario básico de toda la aplicación. En ella es posible crear, abrir o modificar instrumentos. Se pueden visualizar los elementos geométricos del instrumento y editarlos directamente. Para explicar los controles y funciones de esta ventana se utiliza la numeración mostrada en la figura 2.3.

- 1) **Imagen de la boquilla:** Muestra un esquema simplificado de las dimensiones de la boquilla de una flauta dulce. El recuadro negro corresponde a la ventana entre el

canalillo y el bisel, de longitud “W” y anchura “D”, y donde “h” representa la anchura del canalillo.

- 2) **Imagen del resonador:** Visualiza un esquema de los distintos elementos unidos que forman el resonador: tubos cónicos, tubos cilíndricos y agujeros de digitación. Los números de la parte superior muestran la distancia (en cm.) entre cada agujero de digitación y el principio del tubo resonador. En la parte inferior se muestran los números correspondientes a cada elemento.

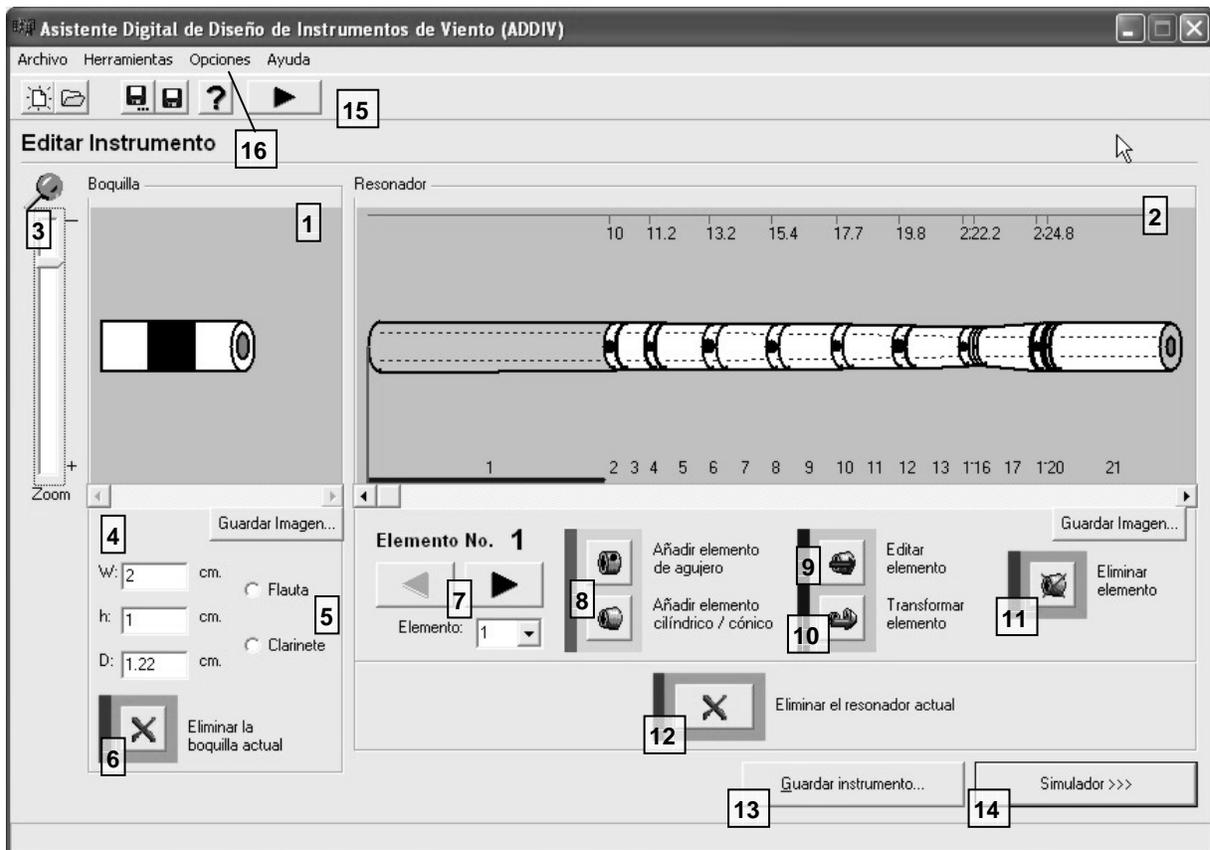


Figura 2.3: Ventana de diseño del instrumento. La numeración de los controles corresponde a la explicación del texto

- 3) **Barra de acercamiento (zoom):** Esta barra permite acercar o alejar simultáneamente las vistas de los controles de imagen de boquilla (1) y resonador (2).
- 4) En estos cuadros de texto se establecen las cantidades “D”, “h” y “W” en centímetros.
- 5) Estos botones de opción excluyente permiten elegir el modelo de boquilla que se va a considerar: de flauta o de clarinete. Actualmente, sólo está habilitado el tipo flauta.
- 6) **Eliminación de la boquilla:** Inicializa los valores de “D”, “h” y “W”.

- 7) **Controles de selección de elemento:** Sirven para definir el elemento seleccionado, el cual podrá ser editado, transformado, borrado, o junto al cual se insertará un nuevo elemento. Los botones de flecha cambian la selección a los elementos contiguos al actual. La lista permite seleccionar directamente el número del elemento deseado. El último número corresponde a un elemento que no existe aún y que puede insertarse al final del tubo resonador.
- 8) **Inserción de elementos:** Permiten añadir elementos nuevos (tubos, agujeros de digitación) a la izquierda del elemento seleccionado. Al ser presionados, abren las ventanas de creación de elementos.
- 9) **Edición de elementos:** Abre la ventana del editor de elementos, que permite cambiar las características geométricas del elemento seleccionado.
- 10) **Transformación de elementos:** Convierte un elemento de agujero de digitación seleccionado en un elemento de tubo y viceversa. Después abre el editor de elementos.
- 11) **Eliminación de elementos:** Borra el elemento seleccionado.
- 12) **Eliminación del resonador:** Borra todos los elementos del resonador.
- 13) **Guardar instrumento:** Guarda el instrumento actual en un archivo “\*.wwi”.
- 14) **Simulador:** Abre la ventana de simulación, una vez que el instrumento ya ha sido creado.
- 15) **Barra de herramientas:** Contiene los botones de manejo de archivos (“Nuevo”, “Abrir”, “Guardar como”, “Guardar” y “Ayuda”, en orden de izquierda a derecha), así como un botón redundante para la apertura del simulador.
- 16) **Menú “Opciones \ Opciones generales”:** Este menú abre la ventana de opciones generales.

### 2.2.3 Ventanas de creación o edición de elementos

Estas ventanas permiten establecer las características físicas de los dos tipos de elementos del resonador: orificios de digitación (figura 2.4) y tubos cónicos o cilíndricos (figura 2.5).

Las imágenes del resonador y de los elementos que muestran las ventanas de diseño y de edición y creación de elementos, son esquemas ideados para mejorar la operación de crear y editar el instrumento elemento por elemento. Para hacer más fácil la edición de los agujeros de digitación (principalmente en configuraciones con orificios muy cercanos), se les

representa acompañados de elementos cilíndricos con la misma longitud que el diámetro del agujero.

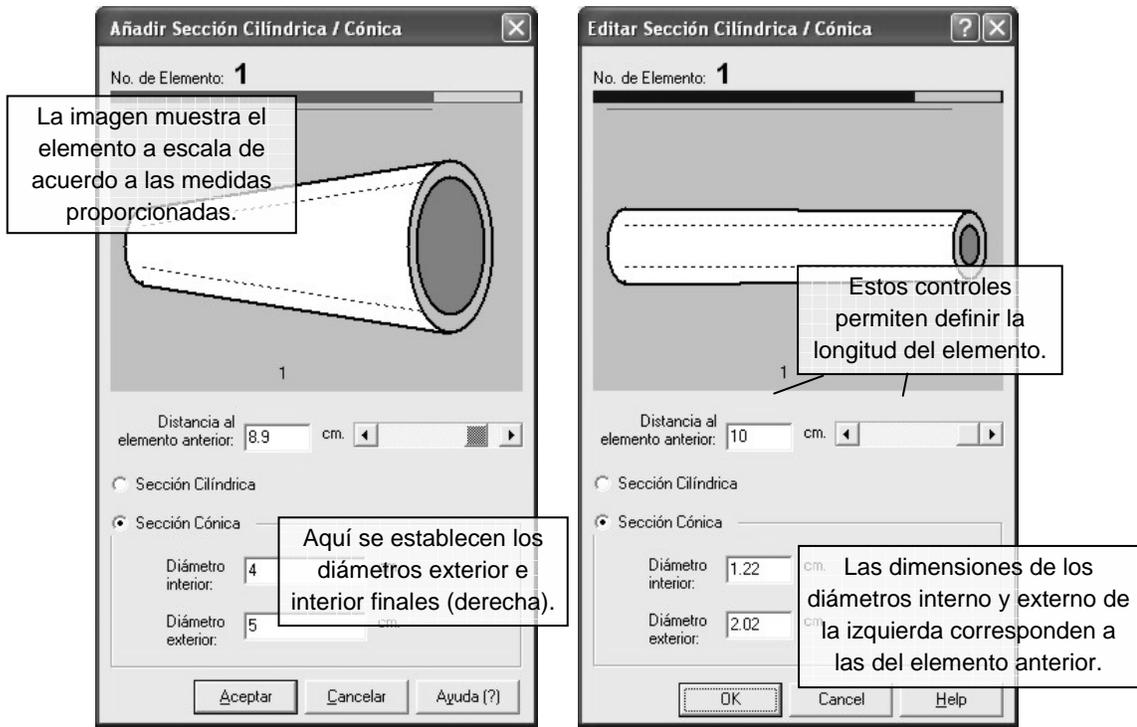


Figura 2.4: Ventanas de creación y edición de elementos cilíndricos o cónicos. Los controles de ambas ventanas son idénticos.

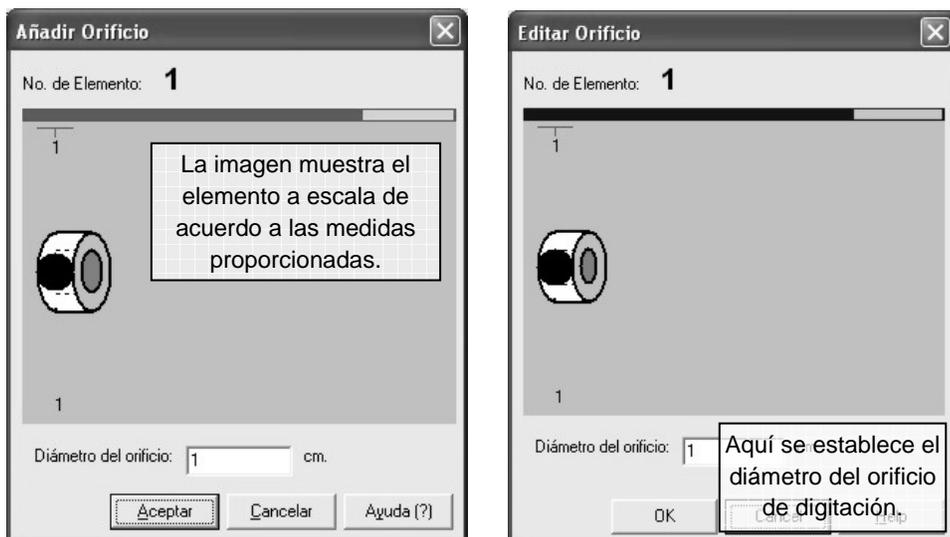
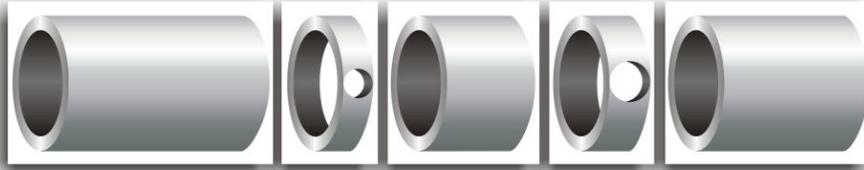


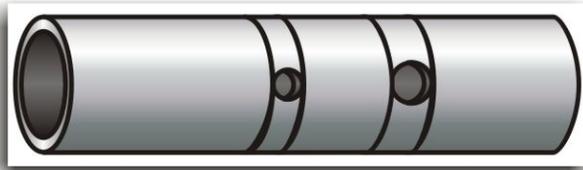
Figura 2.5: Ventanas de creación y edición de elementos de orificio de digitación. Los controles de ambas ventanas son idénticos

Sin embargo, en el modelo de simulación, los orificios son elementos que no tienen longitud. Por ello, los esquemas no deben considerarse como imágenes previas precisas del instrumento (véase la figura 2.6).

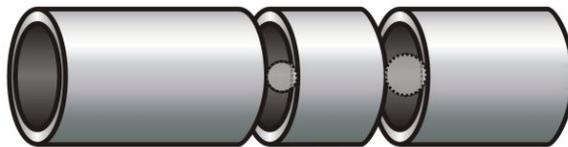
Visualización de elementos separados (ventanas de creación o edición de elementos).



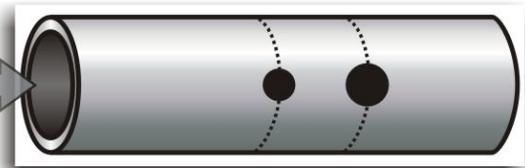
Visualización del resonador (ventana de diseño).



Ajuste.



Instrumento real.



*Figura 2.6: Distintas representaciones gráficas de los elementos del instrumento. Los elementos correspondientes a agujeros de digitación son representados en una sección cilíndrica, dibujada a parte de los elementos circundantes (visualización de elementos separados y de resonador, arriba y centro). Sin embargo, los elementos de orificio tienen longitud 0 y se superponen en las secciones de tubo contiguas (instrumento real, abajo).*

## 2.2.4 Ventana de simulación

En esta ventana se encuentran los controles de generación de sonidos, y los que permiten la visualización de resultados. Todo a partir del instrumento creado en la ventana de diseño. La explicación de los controles y funciones de esta ventana utiliza la numeración mostrada en la figura 2.7.

- 1) **Cuadro de digitación:** Es donde se indica, por medio de casillas de verificación, la digitación de los agujeros del instrumento para la prueba actual. Por la simplicidad del modelo, el orificio final del tubo no puede taparse en la simulación.
- 2) **Controles de presión de entrada:** En estos controles se establece la presión inicial en la entrada de la boquilla, que se tomará en cuenta para realizar la prueba de simulación.

- 3) **Cuadro de texto de tiempo de duración:** En este cuadro se determina la cantidad de segundos que durará el tono de prueba actual.
- 4) **Iniciar sonido:** Este botón ejecuta el proceso de simulación con todos los datos integrados. Con los resultados, genera un archivo temporal de sonido y lo reproduce.
- 5) **Guardar sonido:** Realiza los mismos pasos que el botón (4), pero permite guardar el archivo con un nombre y ubicación personalizados. No reproduce el sonido.

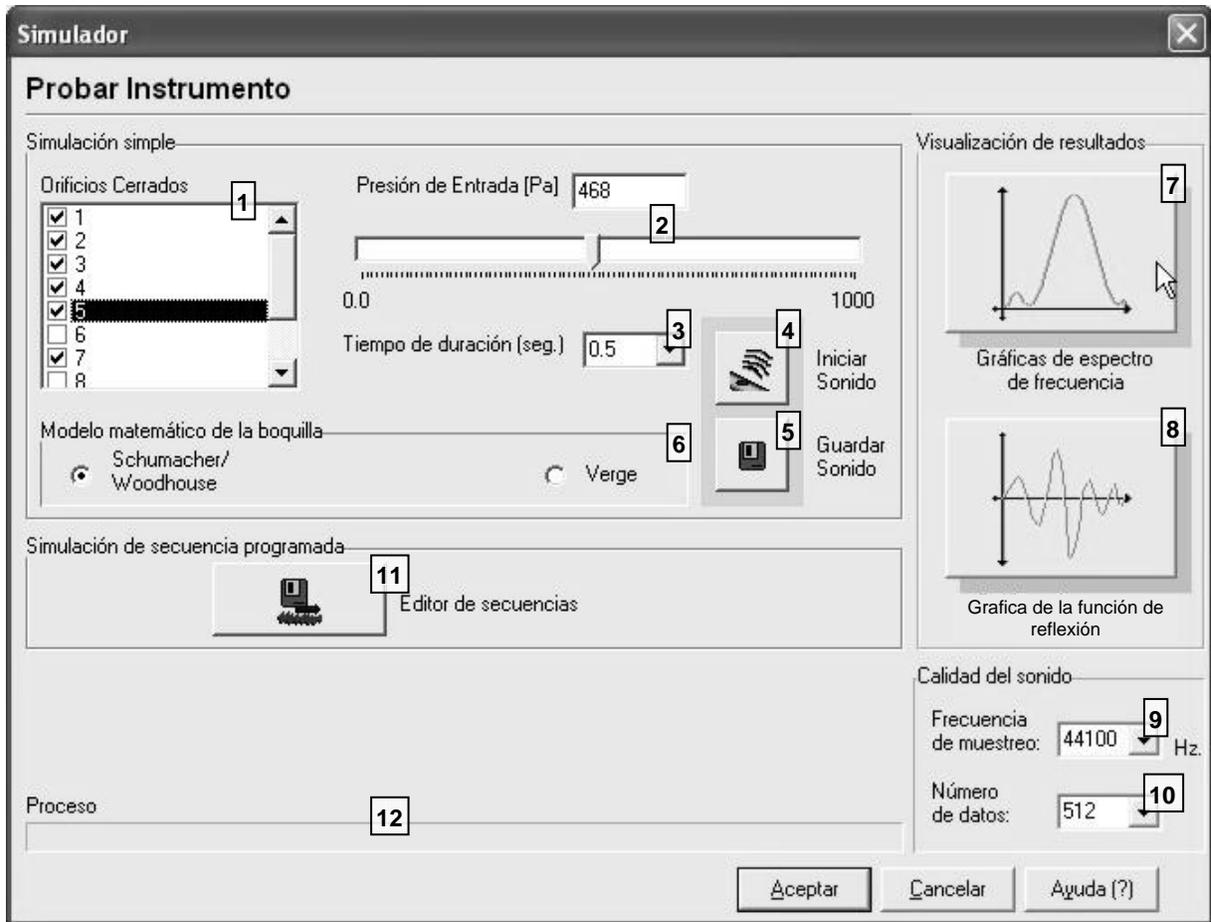


Figura 2.7: Ventana de simulador del instrumento. La numeración de los controles corresponde a la explicación del texto

- 6) Estos controles de opción excluyente permiten escoger el modelo físico matemático de la excitación del instrumento, entre el propuesto por McIntyre et al. [1] y el propuesto por Verge et al. [9]. Esta opción sólo está disponible si se ha escogido previamente para el instrumento, una boquilla de tipo flauta.
- 7) **Gráficas de espectro de frecuencia:** Este botón abre la ventana de gráficas en dominio de la frecuencia.

- 8) **Gráfica de la función de reflexión:** Este botón abre la ventana graficadora de la función de reflexión.
- 9) **Lista de frecuencias de muestreo:** Establece la frecuencia de muestreo digital de la señal sonora que será generada a partir de la simulación.
- 10) **Lista de número de datos:** Controla el número de elementos que contendrá el coeficiente de reflexión, base de la simulación. Por conveniencia computacional, estos números son potencias de 2. El doble de este número corresponde al tamaño de la función de reflexión.
- 11) **Editor de secuencias:** Este botón abre la ventana del editor de secuencias.
- 12) **Barra de progreso de ejecución.** Muestra gráficamente el porcentaje del proceso de ejecución de la prueba, cuando se presionan los botones (4) o (5).

## 2.2.5 Ventanas graficadoras

Estas ventanas permiten observar datos físicos del resonador del instrumento, de manera gráfica. La primera es la ventana de gráficas en dominio de la frecuencia, en la cual se pueden graficar la impedancia acústica de entrada o el coeficiente de reflexión (véase la figura 2.8).

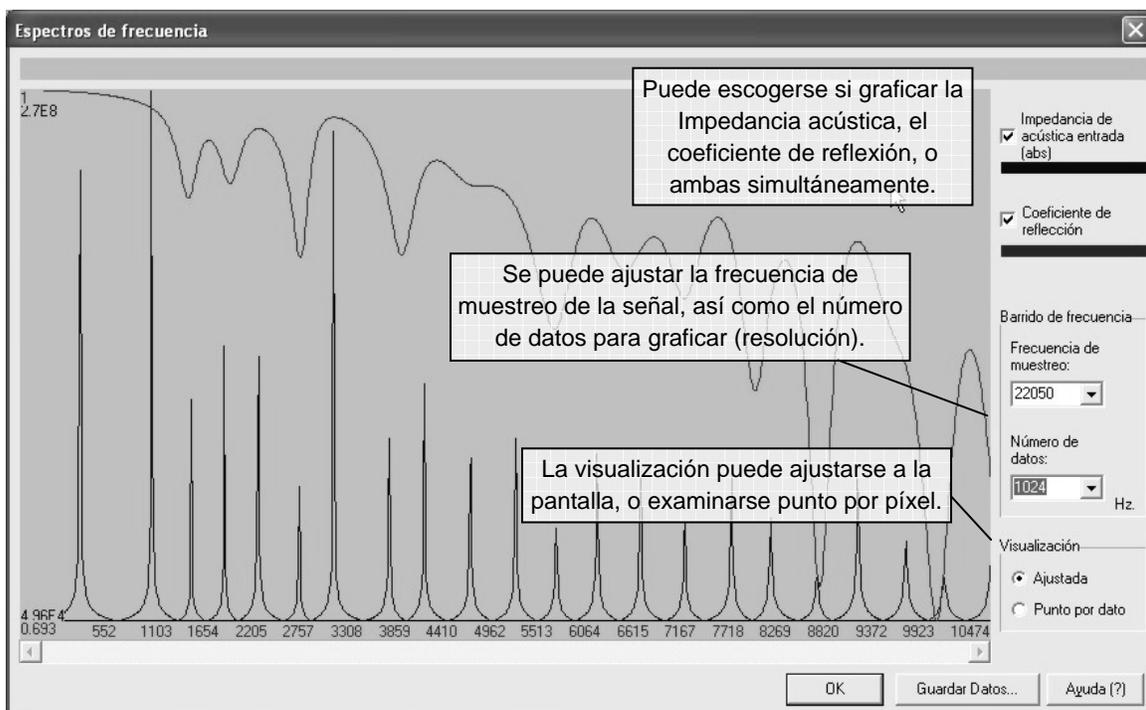


Figura 2.8: Ventana graficadora de espectros en el dominio de la frecuencia.

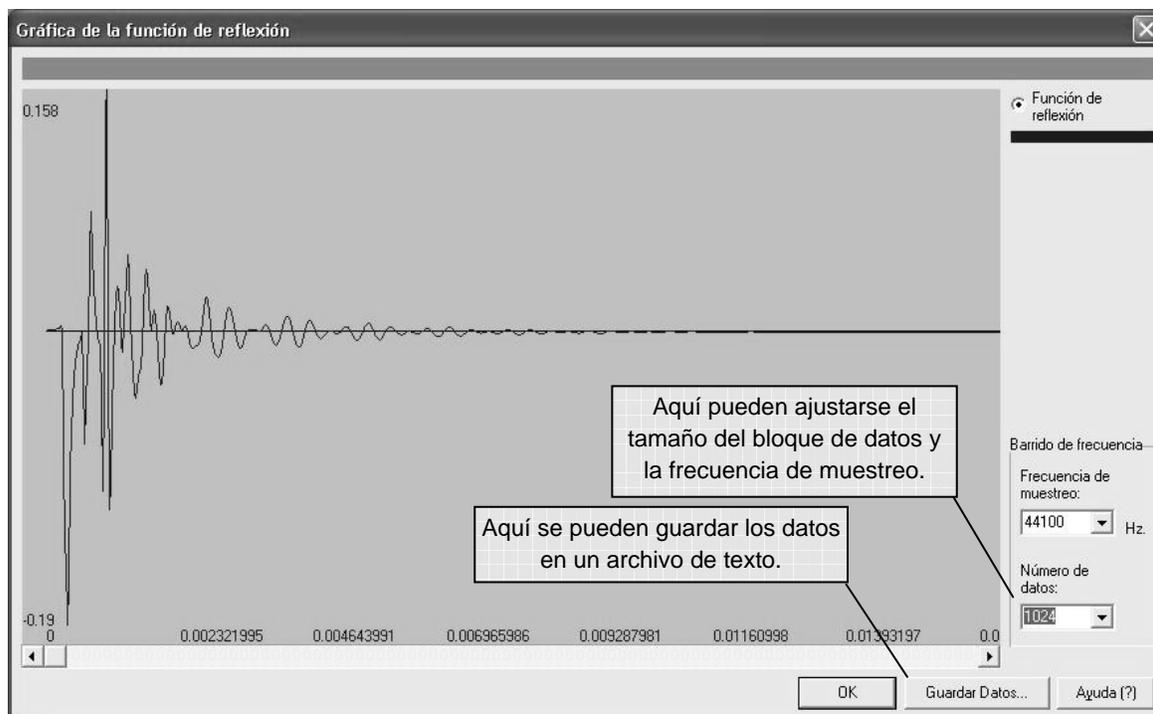


Figura 2.9: Ventana graficadora de la función de reflexión.

La segunda es la ventana de graficadora de la función de reflexión (la cual se encuentra en el dominio del tiempo), en la que se puede observar la función de reflexión calculada para el resonador (véase la figura 2.9).

Ambas ventanas permiten guardar los datos graficados en archivos de datos tabulados.

## 2.2.6 Ventana del editor de secuencias

Esta ventana tiene la utilidad de generar secuencias de interpretación para el instrumento creado. Estas secuencias son varias pruebas simples (como las que se realizan en la ventana de simulación) pero encadenadas. De esta forma, permiten evaluar el comportamiento transitorio del instrumento ante movimientos en la digitación.

Puesto que las características cambiantes son la presión de entrada, los agujeros de digitación cerrados o abiertos, y el tiempo de duración de la nota; son éstas las características que se pueden establecer para cada elemento de la secuencia, como una sucesión de notas (véase la figura 2.10).

Dado que el instrumento creado es arbitrario y puede no presentar un comportamiento musical, los tonos de los elementos de una secuencia no necesariamente tienen correspondencia con notas musicales.

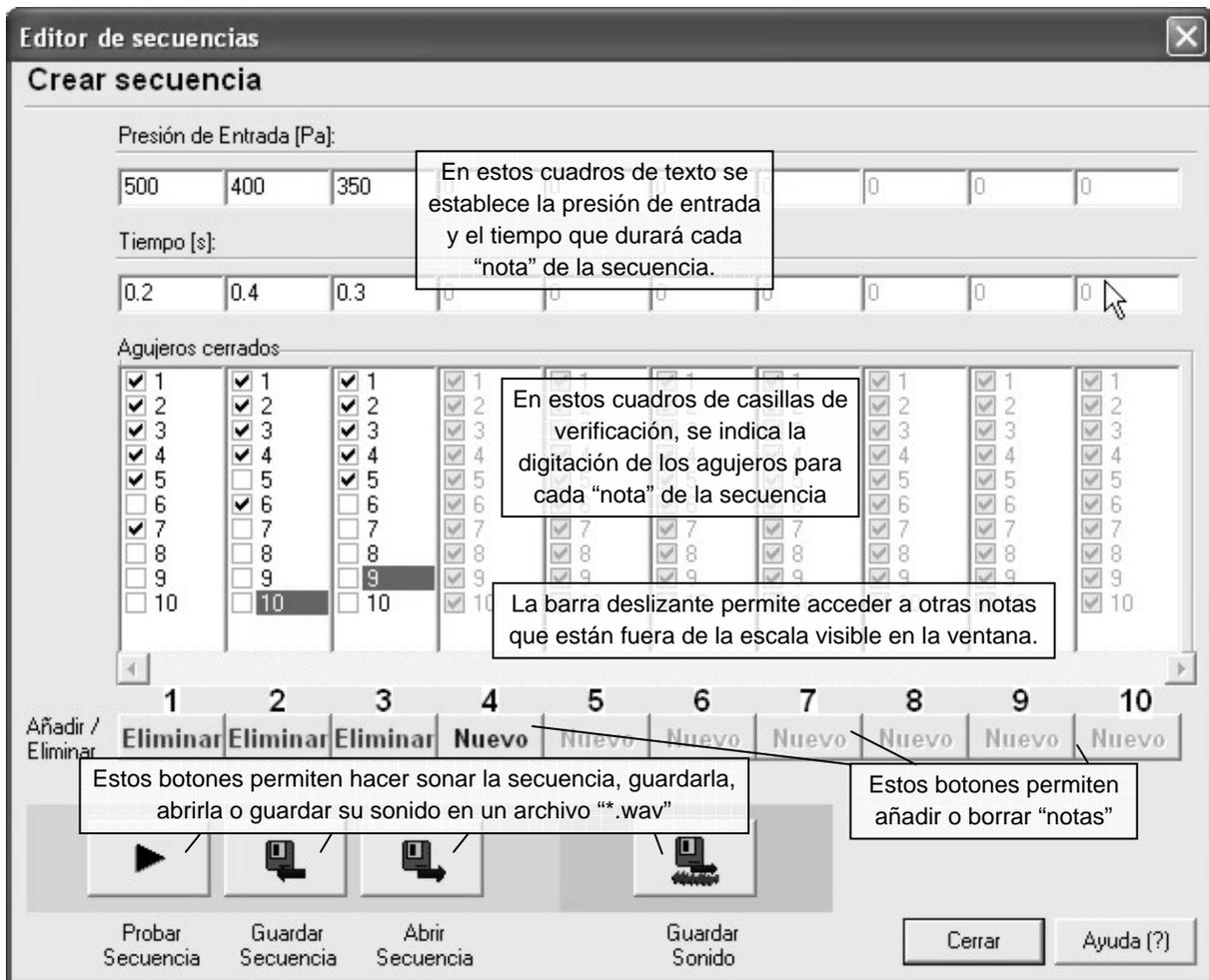


Figura 2.10: Ventana del editor de secuencias.

## 2.2.7 Ventana de opciones generales

La ventana de opciones generales permite establecer las características ambientales que se tomarán en cuenta en las simulaciones, así como la precisión numérica para considerar a dos longitudes geométricas iguales (véase la figura 2.11).

Todas las operaciones dentro del modelo, así como la expresión de mediciones físicas y ambientales en la interfaz, se realizan de acuerdo al sistema decimal MKS, con una excepción. Dadas las reducidas dimensiones espaciales comunes de los instrumentos de viento, las longitudes se manejan en centímetros [cm] en toda la interfaz. Una vez introducidas, estas cantidades son transformadas internamente a metros [m], antes de comenzar las operaciones.

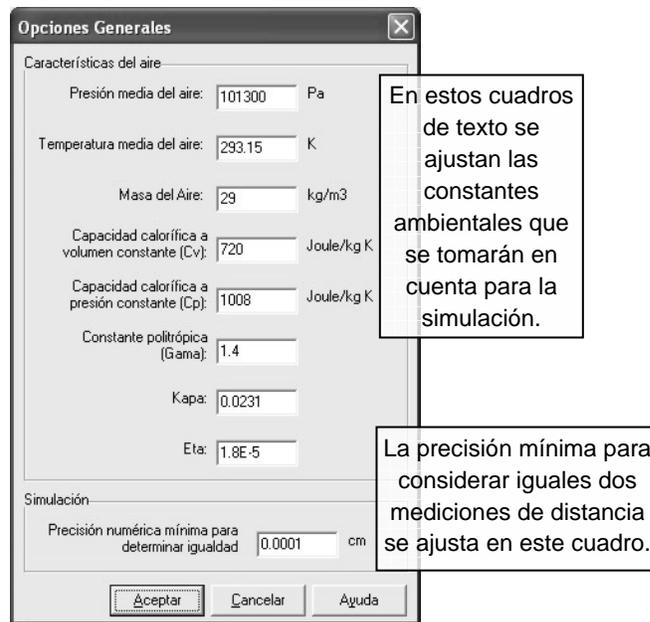


Figura 2.11: Ventana de Opciones Generales del programa.

## 2.3 Bibliotecas de operación

Las bibliotecas de operación son el motor computacional del programa. En ellas se encuentran las funciones que llevan a cabo todas las tareas de la aplicación, las cuales están agrupadas de la siguiente forma:

- Diseño del instrumento (Inst\_Saver.cpp).
- Representación gráfica del instrumento y sus datos (Inst\_Draw.cpp).
- Obtención de la función de reflexión del resonador (Woodwind.cpp).
- Síntesis de sonidos simulados (Sound\_Maker.cpp).

A continuación, se analizarán con más detalle las estructuras de datos y las funciones contenidas en cada una de estas bibliotecas.

### 2.3.1 Biblioteca de estructura de datos del instrumento (Inst\_Saver.cpp)

La biblioteca `Inst_Saver.cpp` contiene las estructuras `INSTRUMENT`, `ELEMENT` y `MOUTH`, que están definidas de la siguiente manera:

**ELEMENT**

Nombre	Tipo (C++)	Descripción
HoleCone <sup>1</sup>	bool	Identificador de elemento, 0 si es tubo y 1 si es agujero de digitación.
CylinderCone	bool	Identificador de tipo de tubo, 0 si es cilíndrico y 1 si es cónico.
IniIntDiam	float	Diámetro interior inicial del elemento.
IniExtDiam	float	Diámetro exterior inicial del elemento.
FinIntDiam	float	Diámetro interior final del elemento.
FinExtDiam	float	Diámetro exterior final del elemento.
Longitude	float	Longitud del elemento.
DiamHole	float	Diámetro del agujero de digitación (equivale a la longitud).
Next	ELEMENT *	Apuntador a una estructura ELEMENT que corresponderá a la siguiente en la lista de elementos. Si el elemento actual es el último, "Next" apunta a NULL [18].

**MOUTH**

Nombre	Tipo (C++)	Descripción
W	float	Distancia entre el final del canalillo y el filo del bisel, longitud de la ventana o "boca".
H	float	Anchura del canalillo
D	float	Anchura de la ventana.

**INSTRUMENT**

Nombre	Tipo (C++)	Descripción
Mouth	MOUTH	Estructura tipo MOUTH, con los datos de la boquilla.
* FirstElement	ELEMENT *	Apuntador a una estructura tipo ELEMENT, que corresponde a la primera de la lista de elementos [17].
NElements	int	Número total de elementos.

<sup>1</sup> En caso de que el elemento sea un agujero de digitación (HoleCone = 0), automáticamente se supone que el elemento es cilíndrico. Los diámetros iniciales y finales del elemento serán idénticos a los diámetros finales del elemento anterior. El único valor que podrá editarse será el diámetro de agujero de digitación (DiamHole). La longitud del elemento (Longitude) será ajustada para coincidir con el valor de DiamHole.

Las funciones de `Inst_Saver.cpp` se encargan de manejar todas las operaciones de estas estructuras de datos:

### **Inst\_Saver.cpp: Principales funciones**

---

**INSTRUMENT\* ConstructNewInstrument()**

Devuelve una nueva estructura INSTRUMENT sin elementos (miembros ELEMENT) de resonador.

---

**AddElement (INSTRUMENT <Inst>, ELEMENT \* <Ele>, int <num>)**

Añade el elemento <Ele> a la estructura <Inst>, en la posición especificada por <num>.

---

**RemoveElement (INSTRUMENT \* <Inst>, int <num>)**

Elimina el elemento en la posición <num> de la estructura <inst>.

---

**EditMouth (INSTRUMENT\* <Inst>, float <W>, float <h>, float <D>)**

Cambia los valores de la estructura MOUTH de <Inst> por los especificados por <W>, <h> y <D>.

---

**SaveInstrument (INSTRUMENT\* <Inst>, char\* <Filename>)**

Guarda la estructura <Inst> en un archivo de datos con nombre y ruta dados por <Filename>

---

**INSTRUMENT \* ConstructOpenInstrument (char\* <Filename>)**

Devuelve una estructura INSTRUMENT que contiene los datos guardados en el archivo <Filename>.

---

**DeleteElements (INSTRUMENT\* <Inst>)**

Elimina todos los elementos de la estructura <Inst>.

---

**ELEMENT \* FindElement (INSTRUMENT\* <Inst>, int <num>)**

Devuelve el apuntador al elemento indicado por <num> de la estructura <Inst>.

---

**int CountElements (INSTRUMENT\* Instrument)**

Devuelve el número de elementos que posee actualmente la estructura <Inst>.

---

**EditElement (INSTRUMENT\* <Inst>, ELEMENT \* <Ele>, int <num>)**

Iguala los valores del elemento <num> de la estructura <Inst> a aquellos contenidos en <Ele>.

---

**bool IsEnoughEqual (float <n1>, float <n2>, float <MinPrec>)**

Devuelve 1 si la diferencia entre <n1> y <n2> es menor a <MinPrec>.

---

**bool IsCompatibleWithNext(ELEMENT \* <Ele>, float <MinPrec>)**

Devuelve 1 si los diámetros finales de <Ele> concuerdan con los iniciales del siguiente elemento.

---

**int CountHoles (INSTRUMENT\* <Inst>)**

Devuelve el número de elementos de tipo agujero de digitación que contiene la estructura <Inst>.

---

**INSTRUMENT \* NewCorrectedInst (INSTRUMENT \* <Inst>)**

Devuelve una estructura INSTRUMENT con la suma de longitudes corregida (véase figura 2.6).

---

**SaveData (double \* <d1>, double \* <d2>, int <N>, double <Min>, double <Max>, char \* <File>)**

Guarda las columnas de datos <d1> y <d2> de <N> elementos en un archivo de datos de nombre y ruta indicados por <File>, tabulados contra un vector lineal que distribuido desde el valor <Min> al valor <Max>.

---

Esta biblioteca está escrita en lenguaje C, y es completamente compatible con cualquier compilador de este lenguaje [19].

### 2.3.2 Biblioteca de objetos gráficos (Inst\_Draw.cpp)

El objetivo de la biblioteca `Inst_Draw.cpp` es generar todos los dibujos y gráficas necesarias para el diseño visual del instrumento y la representación fina de datos. Esta biblioteca está basada en la clase `Canvas` que incluye la Librería de Componentes Visuales (VCL, por sus siglas en inglés) de Borland C++ Builder, y que permite dibujar puntos y figuras geométricas en lienzos reservados dinámicamente [17]. Las principales funciones de esta biblioteca son las siguientes

#### Inst\_Draw.cpp: Principales funciones

---

**Graphics::Tbitmap \* ConstructBitmap(int <X>, int <Y>)**

Devuelve un objeto `Tbitmap`, que es un lienzo de memoria de tamaño <X> por <Y>.

---

**DrawElement (ELEMENT \* <Ele>, TBitmap \* <bitmap>, ...)**

Dibuja un esquema del elemento descrito por la estructura <Ele> en el lienzo <bitmap>.

---

**DrawResonator (INSTRUMENT \* <Inst>, Tbitmap \* <bitmap>, ...)**

Dibuja todos los elementos en secuencia de <Inst> (esquema del resonador), en el lienzo <bitmap>.

---

**DrawMouth (INSTRUMENT\* <Inst>, Tbitmap \* <bitmap>, ...)**

Dibuja un esquema de la estructura `MOUTH` contenida en <Inst>, en el lienzo <bitmap>.

---

**Graphicate (Tbitmap \* <bm>, ... , double \* <Data>, int <N>, ... ,TColor <Color>)**

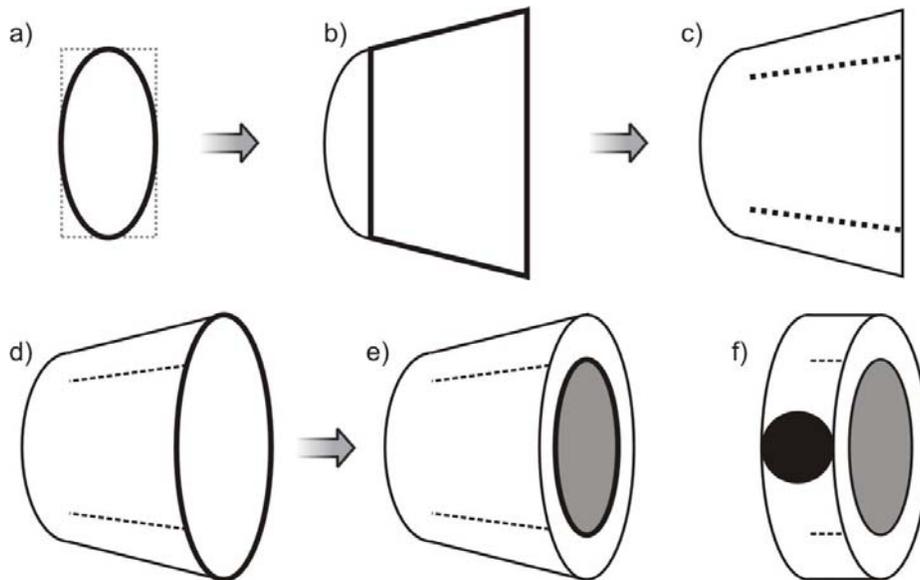
Grafica el vector de datos <Data> de longitud <N> en el lienzo <bm> con líneas de color <Color>.

---

Las representaciones gráficas del programa son básicas para facilitar el proceso de diseño del instrumento. Los dibujos generados son gráficos de dos dimensiones (2-D) con perspectiva isométrica, y se realizan con ayuda de las funciones miembro de la clase `Canvas`.

Estas funciones permiten dibujar líneas, elipses y polígonos en un mapa cartesiano de píxeles. Primero es necesario definir un lienzo de memoria (mapa de bits) sobre el cual se representarán los dibujos. Después, puede comenzar el proceso de dibujo. Es preciso definir la posición y longitudes de las figuras, las cuales son representadas en secuencia. Las figuras creadas posteriormente cubren a las anteriores [17].

Finalmente, el mapa de bits es actualizado y representado en un control visual tipo `Image` de la ventana. La secuencia de pasos para dibujar un elemento del resonador del instrumento puede verse en la figura 2.12:



*Figura 2.12: Secuencia de la representación gráfica de un elemento del resonador. Se inicia con una elipse con el diámetro externo inicial (a). Se dibuja un trapecoide (sección cónica) o un cuadrado (sección cilíndrica) para unir con el diámetro exterior final (b). Se añaden líneas punteadas desde los límites de diámetro interior inicial al los del final (c). Se crea una elipse con el diámetro exterior final (d) y el interior final (e). Si el elemento es un orificio, se representa con una sección cilíndrica, a la cual se añade un círculo oscuro intermedio que representa el agujero de digitación (f).*

Esta biblioteca está escrita en lenguaje C++, y depende estrechamente de la clase `Canvas` de Borland C++ Builder, por lo que no es portátil a otros compiladores [17].

### 2.3.3 Biblioteca del método de matrices de transferencia (`Woodwind.cpp`)

La biblioteca `Woodwind.cpp` utiliza los datos de las estructuras `INSTRUMENT` y `FLUID` para calcular la función de reflexión del resonador del instrumento, por medio de la aplicación computacional del método de matrices de transferencia, detallado en el capítulo anterior.

Tomando en cuenta la conversión de funciones continuas a discretas, discutida en la sección 1.3.7, se define un vector de frecuencias  $F = \{F_1, F_2, F_3, \dots, F_N\}$ , que posee un número  $N$  de elementos y que contiene un barrido de las frecuencias, desde 0 hasta un valor máximo. De acuerdo al teorema de Nyquist [15], el valor máximo corresponderá con la mitad de la frecuencia de muestreo  $f_s$  de la señal, de tal forma que:

$$F_N = \frac{f_s}{2} \quad (2.1)$$

Teniendo en cuenta estas consideraciones, la biblioteca `Woodwind.cpp` define la estructura `FLUID`, y la clase `TMATRIX`. La clase `TMATRIX` representa a la matriz de transferencia, y se integra a partir del vector de frecuencias `F`. Está compuesta de cuatro vectores `A`, `B`, `C` y `D`; cada uno de `N` elementos, de la siguiente forma:

$$T = \begin{bmatrix} A = \{A_1, A_2, A_3, \dots, A_N\} & B = \{B_1, B_2, B_3, \dots, B_N\} \\ C = \{C_1, C_2, C_3, \dots, C_N\} & D = \{D_1, D_2, D_3, \dots, D_N\} \end{bmatrix} \quad (2.2)$$

Los elementos con el mismo índice en cada vector forman la matriz de transferencia de una frecuencia específica, de acuerdo al índice del vector `F`. Por lo tanto, para la frecuencia  $F_n$ , la matriz de transferencia  $T_n$  estará dada por:

$$T_n = \begin{bmatrix} A_n & B_n \\ C_n & D_n \end{bmatrix} \quad (2.3)$$

Los miembros de la clase `TMATRIX` son los siguientes:

TMATRIX		
Variables miembro		
Nombre	Tipo (C++)	Descripción
A	<code>valarray&lt;complex&lt;double&gt;&gt; *<sup>2</sup></code>	Apuntador al vector de elementos A.
B	<code>valarray&lt;complex&lt;double&gt;&gt; *</code>	Apuntador al vector de elementos B.
C	<code>valarray&lt;complex&lt;double&gt;&gt; *</code>	Apuntador al vector de elementos C.
D	<code>valarray&lt;complex&lt;double&gt;&gt; *</code>	Apuntador al vector de elementos D.
Funciones miembro		
Nombre	Descripción	
<code>TMatrix (int &lt;N&gt;)</code>	Constructor de la clase <code>TMATRIX</code> . Crea cuatro vectores de dimensión <code>&lt;N&gt;</code> , a los que apuntan las variables <code>A</code> , <code>B</code> , <code>C</code> y <code>D</code> .	

La estructura `FLUID` contiene las propiedades ambientales que se utilizan en la simulación:

<sup>2</sup> Los vectores de datos tipo `<complex>` y `<valarray>` permiten realizar operaciones aritméticas complejas entre elementos de vectores, con la misma sintaxis que simples valores unitarios [20].

## FLUID

Nombre	Tipo (C++)	Descripción	Valor medio ambiental
P	float	Presión atmosférica [Pa].	101300 Pa
T	float	Temperatura ambiental [K].	293.15 K
mass	float	Masa molar del fluido [kg/mol].	29.0 kg/mol
Cv	float	Calor específico del fluido a volumen constante [J/kg·K].	720 J/kg·K
Cp	float	Capacidad calorífica del fluido a presión constante [J/kg·K].	1008 J/kg·K
gamma	float	Constante politrópica del fluido.	1.4
rho	float	Densidad del fluido [kg/m <sup>3</sup> ].	1.21 kg/m <sup>3</sup>
c	float	Velocidad ambiental del sonido [m/s].	342 m/s
alpha_p	float	Coefficiente de expansión térmica a presión constante [Pa <sup>-1</sup> ].	3.41 x 10 <sup>-3</sup> K <sup>-1</sup>
Kappa_t	float	Compresibilidad volumétrica a temperatura constante [K <sup>-1</sup> ].	9.87 x 10 <sup>-6</sup> Pa <sup>-1</sup>
Kappa	float	Conductividad térmica del aire [W/m·K]	0.0231 W/m·K
eta	float	Viscosidad dinámica del aire [N·s/m <sup>2</sup> ]	0.000018 N·s/m <sup>2</sup>

Las principales funciones de esta biblioteca son las siguientes:

### Woodwind.cpp: Principales funciones

#### **FLUID \* ConstructAir(float <T>, float <P>)**

Devuelve una estructura FLUID, con los valores de temperatura <T>, presión <P> y los demás valores correspondientes a los promedios atmosféricos.

#### **Rcoef (INSTRUMENT \* <Inst>, ..., bool \* <Closed>, FLUID \* <Fluid>, valarray<complex<double> > \* <Z>, valarray<complex<double> > \* <RCoef>)**

Calcula la impedancia acústica característica, el coeficiente de reflexión y los almacena en <Z> y <Rcoef>, respectivamente. El cálculo se realiza con base en los datos del instrumento <Inst>, la digitación <Closed> y el fluido <Fluid>.

#### **double \* RFunc( INSTRUMENT \* <Instrument>, ..., bool \* <Closed>, FLUID \* <Fluid>,...)**

Devuelve el apuntador a un vector que contiene la función de reflexión.

### 2.3.4 Biblioteca de sonido y manejo de secuencias (Sound\_Maker.cpp)

En la biblioteca `Sound_Maker.cpp` se encuentran las funciones que generan y ejecutan los archivos de sonido de simulación. Esta biblioteca representa el último paso en la secuencia de ejecución del programa. También se incluyen las estructuras `NOTE` y `SEQUENCE`, que contiene los datos de una secuencia de interpretación en el instrumento, y que puede ser editada en la ventana del editor de secuencias. Las estructuras están compuestas de la siguiente forma:

NOTE		
Nombre	Tipo (C++)	Descripción
P	float	Presión de entrada constante para la nota, en Pascales.
Time	float	Tiempo que durará el tono generado, en segundos.
ClosedHoles	bool *	Apuntador a un arreglo de elementos de tipo booleano. El número de elementos del arreglo corresponderá a la cantidad de agujeros de digitación del instrumento. Un elemento 0 significará un agujero abierto. Un elemento 1 corresponderá a un agujero cerrado.

SEQUENCE		
Nombre	Tipo (C++)	Descripción
NHoles	int	Numero de orificios de digitación de los que consta el instrumento actual.
Notes[n]	NOTE	Arreglo de longitud "n" cuyos elementos son estructuras de tipo NOTE, y que corresponden a las distintas notas de la secuencia. El valor "n" es mayor o igual a la cantidad de notas.

Las principales funciones de esta biblioteca son:

#### Sound\_Maker.cpp: Funciones de uso externo

**double \* SchWoodModel (INSTRUMENT \* <Inst>, double \* <R>, ..., FLUID \* <F>, double <P>,...)**

Devuelve un vector con los datos de sonido simulado con el modelo de flauta de McIntyre et al. [1], utilizando un vector con la función de reflexión <R>, la estructura de datos del instrumento <Inst>, el ambiente definido por <F> y la presión de entrada <P>.

**double \* SchWoodClarinetModel (INSTRUMENT \* <Inst>, double \* <R>, ..., FLUID \* <F>, double <P>,...)**

Devuelve un vector con los datos de sonido simulado con el modelo de clarinete de McIntyre et al. [1], utilizando un vector con la función de reflexión <R>, la estructura de datos del instrumento <Inst>, el ambiente definido por <F> y la presión de entrada <P>.

---

**double \* VergeModel (INSTRUMENT \* <Inst>, double \* <R>, ..., FLUID \* <F>, double <P>,...)**

Devuelve un vector con los datos de sonido simulado con el modelo de flauta de Verge et al. [9], utilizando un vector con la función de reflexión <R>, la estructura de datos del instrumento <Inst>, el ambiente definido por <F> y la presión de entrada <P>.

---

**double \* MakeSound(INSTRUMENT \* <Inst>, FLUID \* <Fluid>, bool \* <Closed>, ...)**

Devuelve un vector de datos con el sonido que deberá convertirse en un archivo de audio.

---

**MakeWav (char \* <File>, double \* <data>, int <fs>, float <time>, ...)**

Crea un archivo \*wav con nombre y ruta indicados por <File>, con los datos del vector <data>, a una frecuencia de muestreo <fs>, por una duración de <time> en segundos.

---

**StartSequence(SEQUENCE \* <Seq>, int <NHoles>)**

Inicializa la estructura de secuencia <Seq> con el valor de NHoles igual a <NHoles> y con los apuntadores ClosedHoles de las estructuras NOTE de <Seq> igualadas a NULL.

---

**AddNewNote(SEQUENCE \* <Seq>)**

Añade una nueva estructura NOTE al final de la secuencia <Seq>, con valores predefinidos.

---

**SetNote(SEQUENCE \* <Seq>, NOTE \* <Note>, int <n>)**

Copia los valores de <Note> en la nota número <n> de la estructura <Seq>.

---

**DeleteNote(SEQUENCE \* <Seq>, int <n>)**

Elimina los valores de la nota <n> de la secuencia <Seq> y recorre los valores siguientes.

---

**CountNotes(SEQUENCE \* <Seq>)**

Cuenta las notas (estructuras NOTE) contenidas en la estructura <Seq>.

---

**DeleteSequence (SEQUENCE \* <Seq>)**

Borra todos los datos contenidos en la estructura <Seq>.

---

**OpenSequence (SEQUENCE \* <Seq>, INSTRUMENT \* <Inst>, char \* <file>)**

Llena la estructura <Seq> con los datos almacenados en el archivo de nombre y ruta <file>, siempre y cuando correspondan con el número de agujeros de digitación del instrumento <Inst>.

---

**SaveSequence (SEQUENCE \* <Seq>, INSTRUMENT \* <Inst>, char \* <file>)**

Guarda los datos de la estructura <Seq> y el número de agujeros de digitación del instrumento <Inst>, en el archivo de nombre y ruta <file>.

---

**MakeSoundSequence(SEQUENCE \* <Seq>, INSTRUMENT \* <Inst>, ..., char \* <file>)**

Crea un archivo de sonido con el instrumento <Inst> y la secuencia <Seq>, y la guarda en el archivo y ruta <file>.

---

## 2.4 Algoritmos principales

### 2.4.1 Algoritmo de cálculo de la función de reflexión

La primera parte del proceso de simulación consiste en generar el vector de datos de la función de reflexión, que representará en el modelo al resonador del instrumento.

Las entradas de este algoritmo son una estructura <Inst> de tipo INSTRUMENT (que contiene los datos del resonador del instrumento), una estructura <Air> de tipo FLUID, un vector <Dig> de tipo booleano, en el que se especifica el estado de los orificios del instrumento (FALSE = abierto, TRUE = cerrado), la frecuencia de muestreo <fs> y el número de datos <N> que compondrán el vector de frecuencias. Para mejorar la eficiencia de la función  $\text{ifft}[\ ]$ , la cual es utilizada al final del algoritmo, el valor de <N> debe ser igual a alguna potencia de 2.

- El primer paso consiste en generar un vector <F> de <N> elementos, que contenga un barrido de todas las frecuencias, desde 0, hasta la mayor posible de acuerdo a la función de reflexión <fs>. De acuerdo al teorema de Nyquist [15], esta frecuencia corresponde a la mitad de <fs>.
- Se define la estructura <T> de tipo TMATRIX, con <N> elementos.
- Si el número <Num> de elementos del resonador es igual a cero, no puede calcularse propiamente una función de reflexión, así que se termina el proceso. De lo contrario, se obtienen los valores constantes para el proceso: los números de onda <k>, las impedancias acústicas características de entrada <Zo> y las de radiación <Zr> para cada frecuencia de <F>.
- Se analiza cada elemento del resonador, desde 1 hasta <Num>. Se determina qué tipo de elemento es (cónico, cilíndrico o de orificio) y se resuelven las ecuaciones correspondientes para obtener la matriz de transferencia correspondiente. En el caso de los elementos de tipo orificio, se verifica en el vector <Dig> si el orificio está cerrado o abierto. La matriz obtenida se multiplica con las calculadas anteriormente, hasta el último elemento.
- Con los valores obtenidos, se calcula vector del coeficiente de reflexión <RC>.
- El vector de datos <RC> se prepara y se le aplica la transformada rápida de Fourier, para obtener el vector de datos de la función de reflexión <RF> (véase la figura 2.14).

Este algoritmo puede observarse con mayor claridad en la figura 2.13.

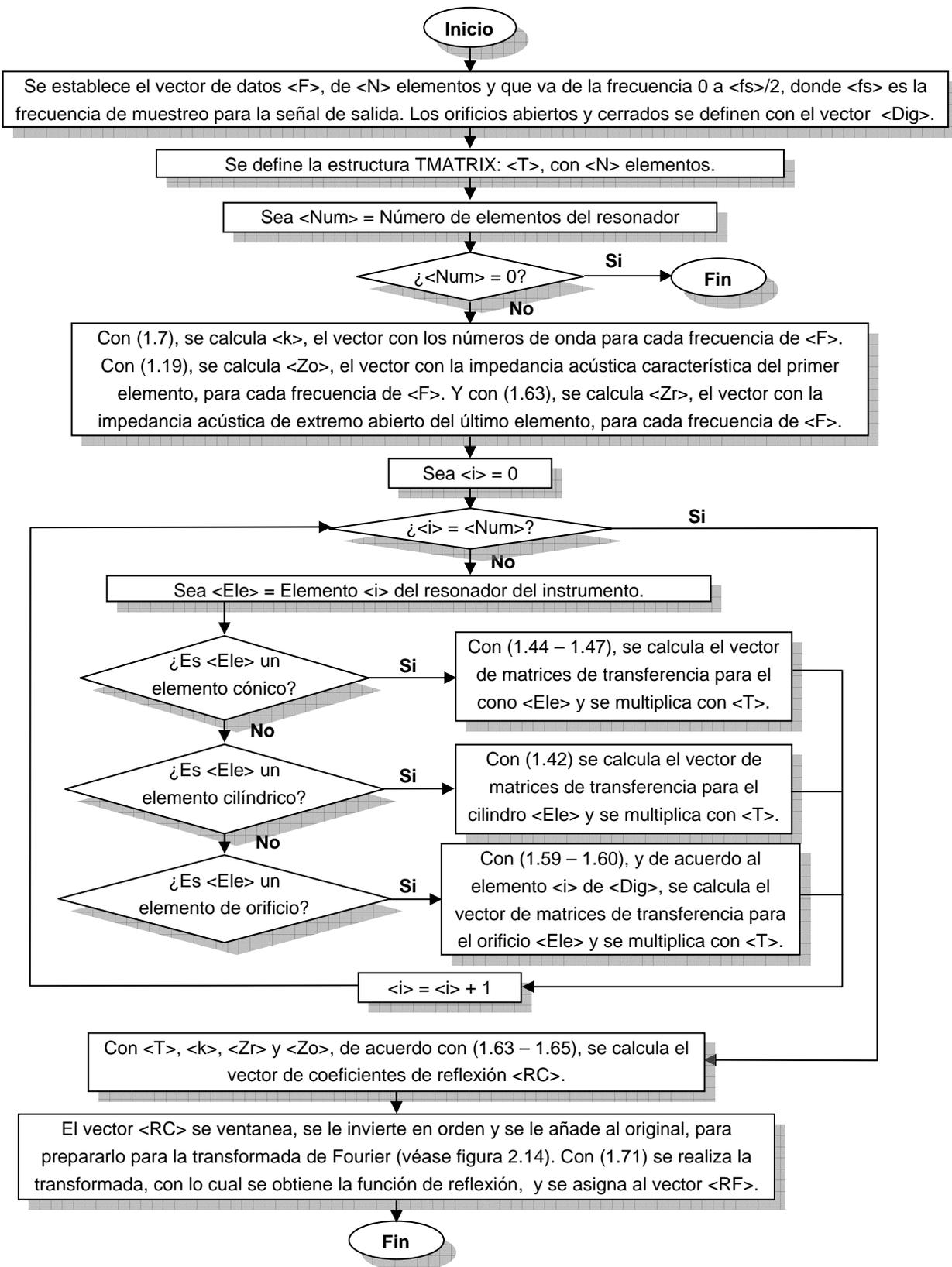


Figura 2.13: Diagrama de flujo del algoritmo de cálculo de la función de reflexión.

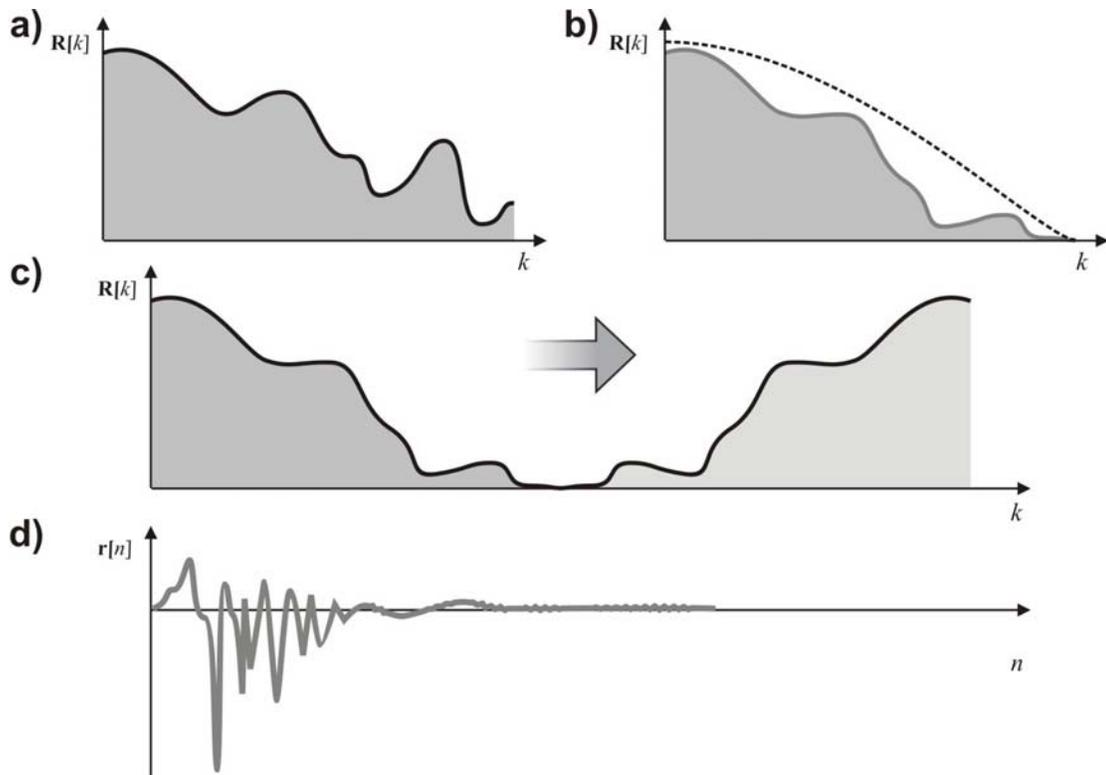


Figura 2.14: Pasos para el cálculo de la función de reflexión a partir del coeficiente de reflexión: a) Gráfica de la parte real del coeficiente de reflexión, en dominio de la frecuencia. b) El coeficiente es “ventaneado” con una semi ventana Hanning, para garantizar su atenuación hasta cero en el límite de alta frecuencia. c) Se duplican los datos en modo espejo, para cumplir con la simetría compleja conjugada. d) Después de aplicar la transformada inversa de Fourier, se obtiene la función de reflexión en dominio del tiempo.

#### 2.4.2 Algoritmo de simulación de sonido: McIntyre et al.

La siguiente etapa consiste en utilizar la función de reflexión en el modelo del oscilador, para simular el sonido. Se implementaron dos modelos, el propuesto por McIntyre et al. [1] y el propuesto por Verge et al. [9] de una flauta dulce.

Las entradas de este algoritmo son la estructura de datos <Inst> de tipo INSTRUMENT del instrumento, la función de reflexión <RF> obtenido del resonador del instrumento (de tamaño <N>), el tiempo de duración del sonido <Time> en segundos, la frecuencia de muestreo <fs> y la presión de entrada <P>. El modelo fue implementado para realizar los siguientes pasos.

- Se calculan los valores de la impedancia acústica característica <Zo>, la velocidad del flujo de entrada <Uj> y el retardo <tau>.
- Se inicializan dos vectores dinámicos de presión acústica <p> y de flujo volumétrico <f>. Se inicializan sus primeros <N> valores a cero. Se calcula el número <Ntotal> de muestras, multiplicando el tiempo de sonido <Time> por la frecuencia de muestreo

<fs> y añadiendo una cantidad <N> para dar espacio al efecto de decaimiento final del tono.

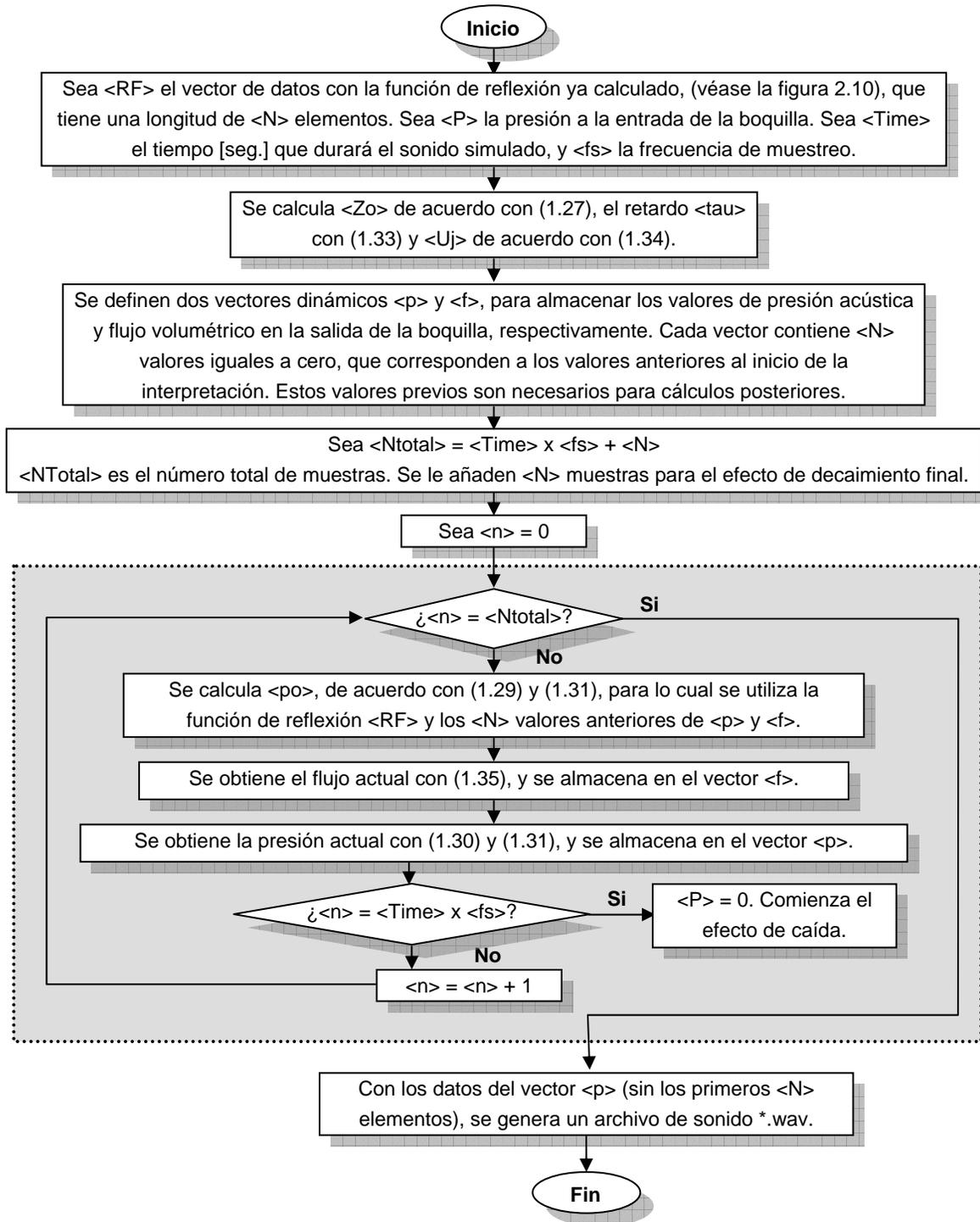


Figura 2.15: Diagrama de flujo del algoritmo de simulación de sonido, de acuerdo a la propuesta de McIntyre et al. [1]. El recuadro punteado indica la sección que es distinta para el algoritmo de Verge et al. [9].

- Inicia el cálculo de cada muestra, desde 0 hasta  $\langle N_{total} \rangle$ . Para cada una:
  - Se calcula el valor actual de la presión de retorno (reflejada) del resonador  $\langle p_o \rangle$ , utilizando las expresiones (1.29) y (1.31). Para realizar la convolución de (1.29), se utilizan los  $\langle N \rangle$  valores anteriores de  $\langle p \rangle$  y  $\langle f \rangle$ .
  - Con la expresión (1.35) se calcula el valor actual del flujo  $\langle f \rangle$ , y se almacena en el vector.
  - Con los valores de  $\langle p_o \rangle$  y  $\langle f \rangle$ , y con (1.30) y (1.31), se calcula el valor actual de la presión total  $\langle p \rangle$ , y se almacena en el vector.
  - Los pasos anteriores se repiten hasta haber llegado a  $\langle N_{total} \rangle = \langle Time \rangle \times \langle fs \rangle$ . En ese momento, la presión de entrada en la boquilla  $\langle P \rangle$  y, por lo tanto, la velocidad del flujo  $\langle U_j \rangle$ , son cero. De esta forma, la excitación se detiene y las oscilaciones decaen paulatinamente en las siguientes  $\langle N \rangle$  muestras. Este efecto de decaimiento evita el corte abrupto de la simulación, y el indeseable “clic” acústico que provoca al final.
- Los elementos almacenados en  $\langle p \rangle$  son normalizados y extraídos para generar el archivo de sonido (excepto los  $\langle N \rangle$  primeros, que son cero).
- Se genera un archivo de audio de Windows (\*.wav).

Este algoritmo puede observarse con mayor claridad en la figura 2.15.

### 2.4.3 Algoritmo de simulación de sonido: Verge et al.

La programación del modelo de Verge es muy similar a la del modelo de McIntyre. Las diferencias radican en el cálculo del flujo volumétrico  $\langle f \rangle$  y la presión acústica  $\langle p \rangle$ , los cuales se obtienen de manera distinta. Los pasos de ejecución de este algoritmo son los siguientes.

- Se obtienen las constantes necesarias para los cálculos del modelo (véase la figura 2.13 y la sección 1.3.2).
- Aparte de  $\langle p \rangle$  y  $\langle f \rangle$ , se declaran tres vectores dinámicos más. Las variables incógnitas de la expresión (1.36) son la presión de entrada al resonador  $\langle p_{in} \rangle$  y la presión de retorno del resonador  $\langle p_o \rangle$ . Puesto que el algoritmo requiere valores anteriores de estas variables para resolver la ecuación diferencial, es conveniente el almacenamiento de los valores pasados de estas variables en vectores de datos. También se define el vector  $\langle x \rangle$ , que almacena datos pasados del primer factor de la expresión (1.39).

- Se calcula, en número de muestras, el tiempo de la operación de retardo de la expresión (1.39), y se almacena en  $\langle \tau \rangle$ .

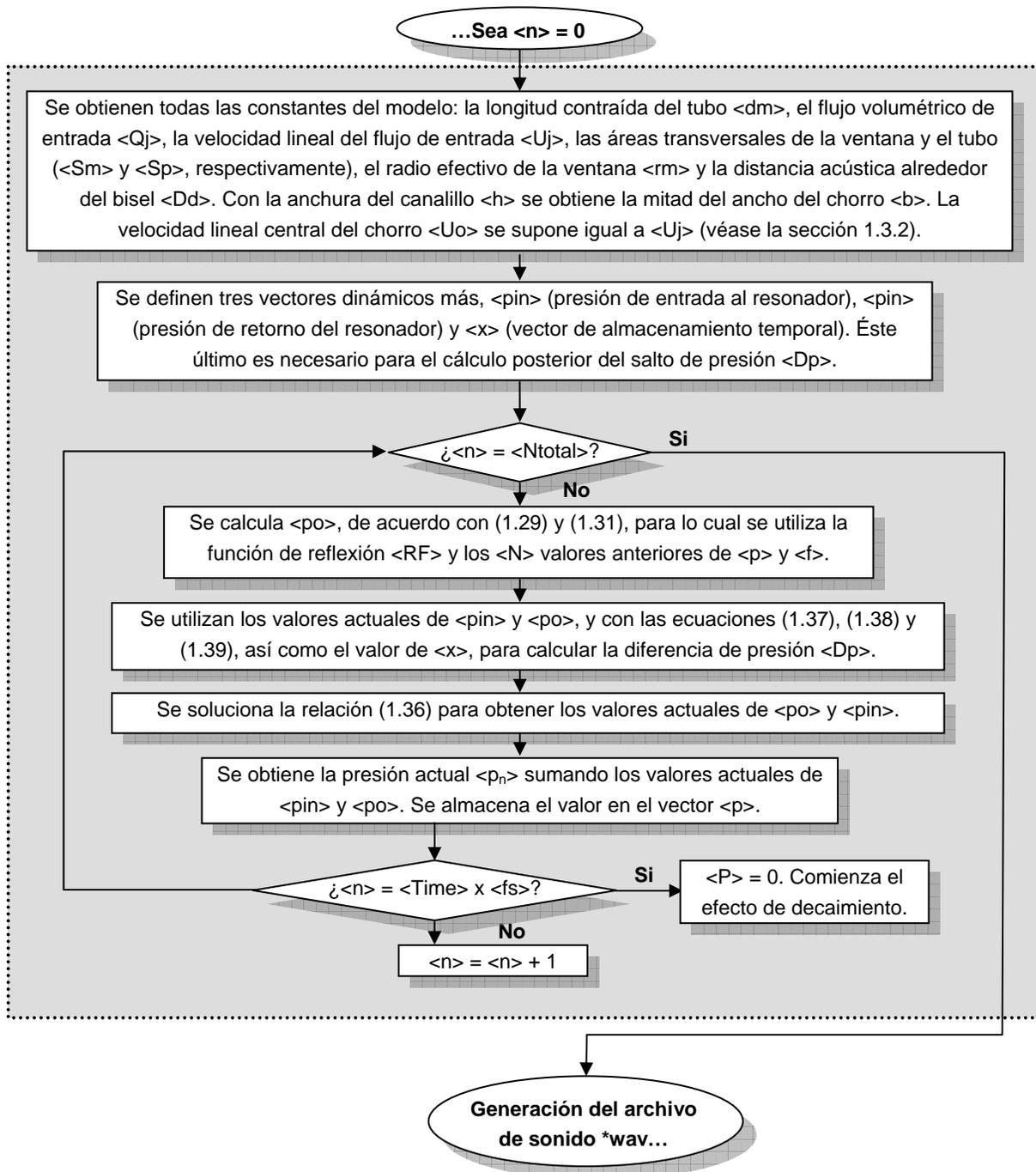


Figura 2.16: Diagrama de la sección del algoritmo de simulación de sonido correspondiente al modelo de Verge et al [9] en el diagrama de la figura 2.12.

- Inicia el cálculo de cada muestra, desde 0 hasta  $\langle N_{total} \rangle$ . Para cada una:
  - Se calcula el valor actual de la presión de entrada del resonador  $\langle p_{in} \rangle$ , utilizando las expresiones (1.29) y (1.31). Para realizar la convolución de (1.29), se utilizan los  $\langle N \rangle$  valores anteriores de  $\langle p \rangle$  y  $\langle f \rangle$ .
  - Se preparan los cálculos del salto de presión  $\langle D_p \rangle$ . La semi-longitud de la anchura del chorro  $\langle b \rangle$  se iguala a la mitad de  $\langle h \rangle$ , la anchura del canalillo. La velocidad lineal central del chorro  $\langle U_o \rangle$  se considera igual a la velocidad lineal del flujo de aire al salir del canalillo  $\langle U_j \rangle$ . El desplazamiento  $\langle Y_o \rangle$  entre el centro del canalillo y el filo del bisel se considera nulo. Para la primera muestra, el desplazamiento del chorro frente al bisel  $\langle \eta \rangle$  (1.39) se inicializa a cero.
  - Se calcula la fuente de presión virtual  $\langle Q_1 \rangle$ , aplicando la expresión (1.38).
  - Se obtiene la presión total de entrada al resonador  $\langle Q_p \rangle$  con (1.40).
  - Se calcula el factor (1) de la expresión (1.39) y se almacena en el vector  $\langle x \rangle$ .
  - Con el valor actual de  $\langle x \rangle$  y aquél obtenido  $\langle \tau \rangle$  muestras atrás, se calcula el valor de  $\langle \eta \rangle$ , de acuerdo con (1.39).
  - Se obtiene las diferenciales de la variable  $\langle Q_j \rangle$ , con el método de Euler [21].
  - Se resuelve numéricamente la ecuación diferencial (1.36) para obtener el valor actual de  $\langle p_o \rangle$ . Se hace uso del método de Runge-Kutta de 4° orden para aproximar las diferenciales [21] [22].
  - Se obtiene el valor de actual de  $\langle p \rangle$ , sumando  $\langle p_{in} \rangle$  y  $\langle p_o \rangle$ . El valor se almacena en el vector dinámico de  $\langle p \rangle$ .
- Una vez terminadas todas las muestras, se procede a la generación del archivo de sonido, a partir de los datos almacenados en  $\langle p \rangle$ , de la misma forma que en el algoritmo de McIntyre.

## **2.5 Prototipos de otros modelos**

En el proceso de adaptación computacional de los modelos de instrumentos acústicos, se experimentó con otras variantes que no fueron implementadas en la versión final del programa, pero que fueron probadas y arrojaron resultados preliminarmente satisfactorios. Estos dos modelos se encuentran documentados también en el trabajo de McIntyre et al. [1], y corresponden a un clarinete y a una cuerda tensa frotada (como la de un violín).

### 2.5.1 El clarinete

El modelo del clarinete, como instrumento de viento, está implementado de la misma forma que el de la flauta dulce. La principal diferencia consiste en la relación no lineal del excitador  $f(p)$ , que tiene un comportamiento aproximado al que puede observarse en la figura 2.16:

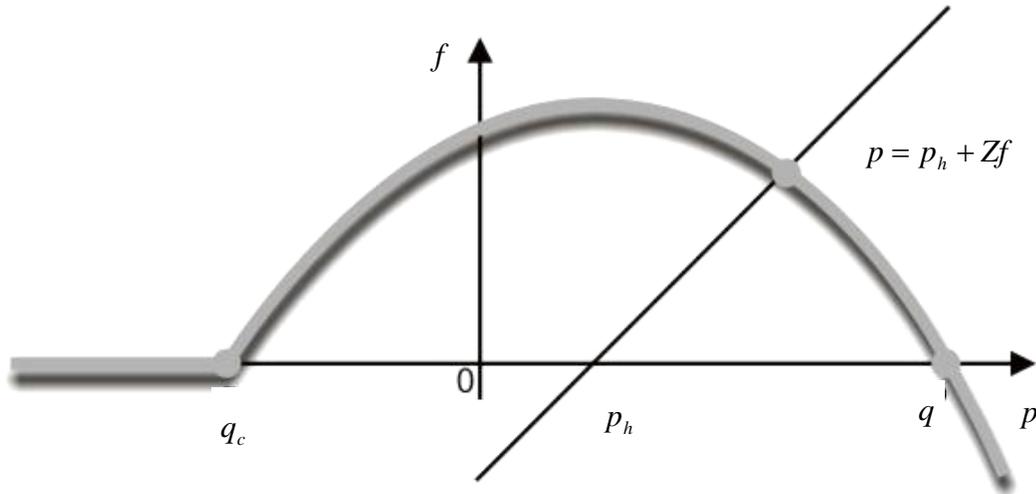


Figura 2.16: Relación no lineal simplificada de  $f(p)$  para un clarinete. Debe resolverse la intersección entre la recta de  $p$  y la curva  $f(p)$  para conocer los valores instantáneos  $f$  de y  $p$ .

Esta relación puede ser aproximada analíticamente con la siguiente expresión [1]:

$$f(p) = k(q - p)(p - q_c) \quad (2.4)$$

Donde  $q$ ,  $q_c$  y  $k$  son parámetros fijos del modelo.

La relación excitación – resonador se describe con las mismas expresiones del modelo de la flauta dulce (1.29 – 1.31). En este caso, la función de reflexión utilizada fue una simple curva gaussiana negativa, que se aproxima a la función correspondiente a un tubo simple abierto.

Este modelo fue programado en MATLAB y un resultado de su implementación puede verse en el gráfico de la figura 2.17.

### 2.5.2 La cuerda tensa frotada

En la cuerda tensa frotada, las variables cambian de significado, pero las relaciones (1.29 – 1.31) se mantienen idénticas. En este caso, la variable  $p$  es la velocidad con la que el arco se mueve sobre la cuerda,  $f$  representa la fuerza de fricción ejercida sobre la cuerda y  $Z$  es ahora la admitancia mecánica de la cuerda.

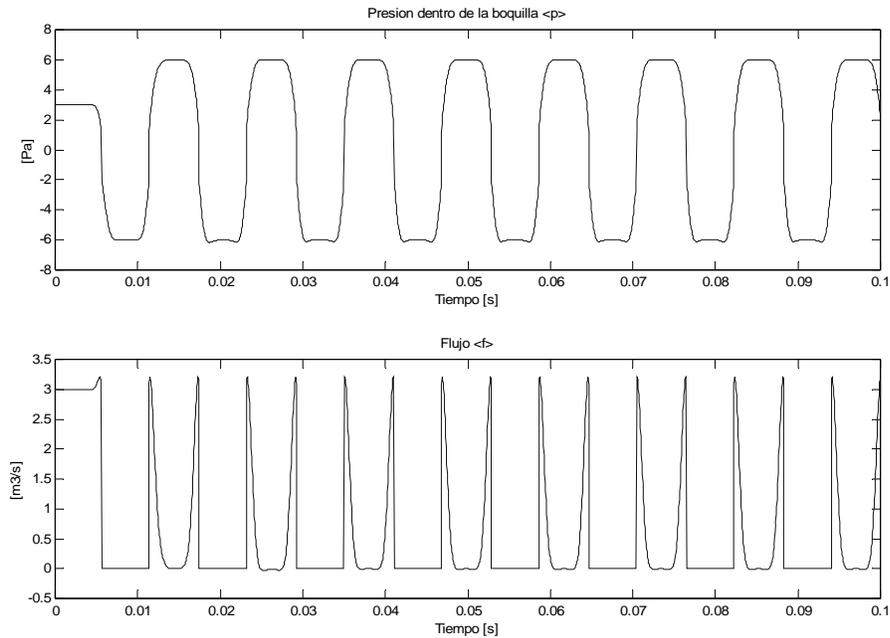


Figura 2.17: Forma de onda obtenida con el programa de simulación de un clarinete, de acuerdo al modelo propuesto por McIntyre et al., con parámetros típicos de un instrumento real.

La ecuación (2.4) debe ser reemplazada por la curva no lineal característica de la cuerda frotada. Tal curva corresponde a la mostrada en la figura 2.18.

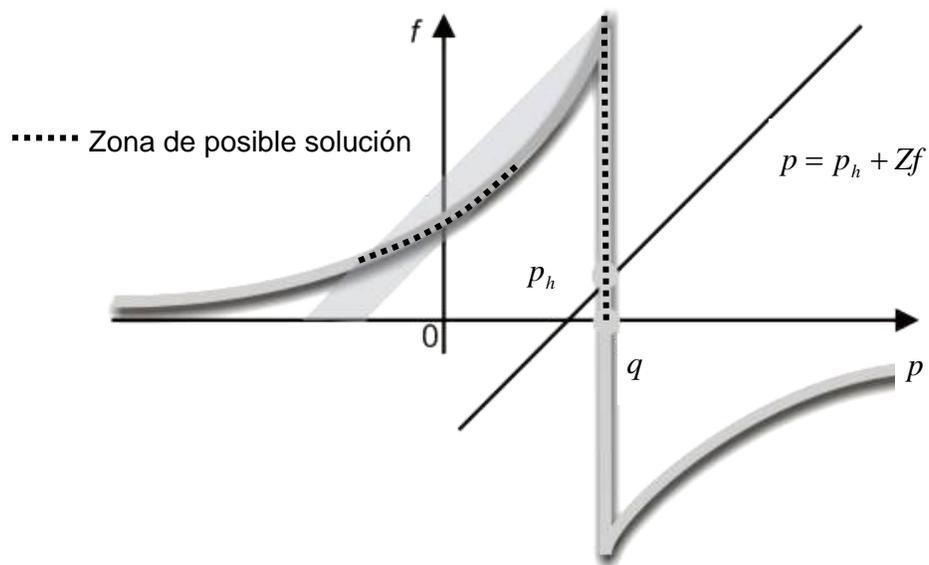
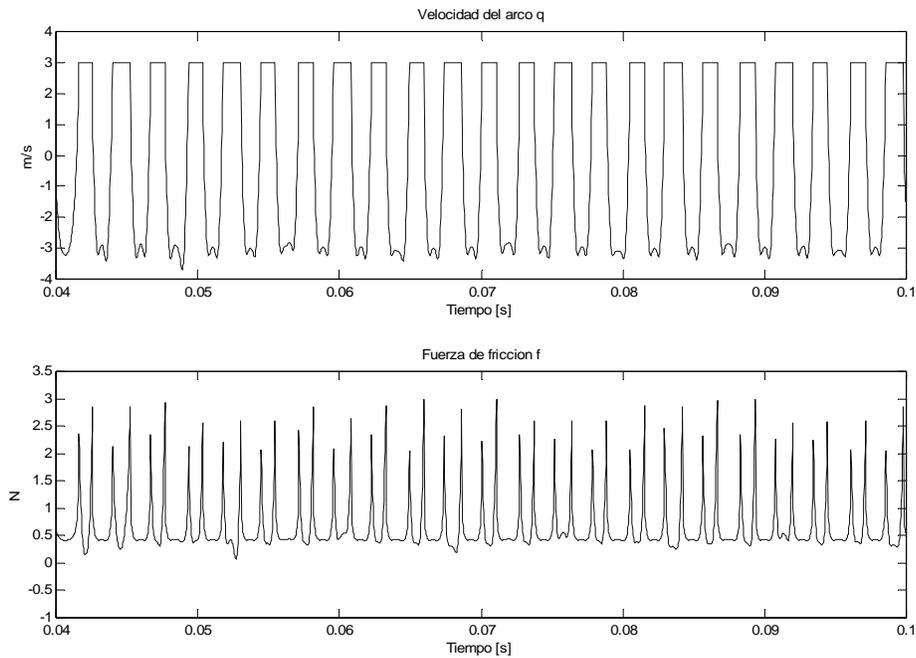


Figura 2.18: Relación no lineal simplificada de  $f(p)$  para una cuerda frotada. Debe resolverse la intersección entre la recta de  $p$  y la curva  $f(p)$  para conocer los valores instantáneos de  $f$  y  $p$ .

Para modelar este comportamiento, se utilizó una aproximación asintótica para valores anteriores a  $p = q$ , con la siguiente expresión [1]:

$$f = -K / (p - q - 1) \quad (2.5)$$

Donde  $K$  y  $q$  son parámetros positivos del modelo. La solución correcta, cuando hay más de una, es aquella más cercana a del estado anterior del sistema, entre las dos zonas de posible solución mostradas en la figura 2.18. Un resultado de la implementación de este modelo puede verse en el gráfico de la figura 2.19



*Figura 2.19: Forma de onda obtenida con el programa de simulación de una cuerda frotada, de acuerdo al modelo propuesto por McIntyre et al., con parámetros típicos de un instrumento real.*

## Capítulo 3: Resultados

### 3.1 Pruebas comparativas

El primer paso en el desarrollo de las pruebas fue la introducción de los datos de un instrumento real al programa.

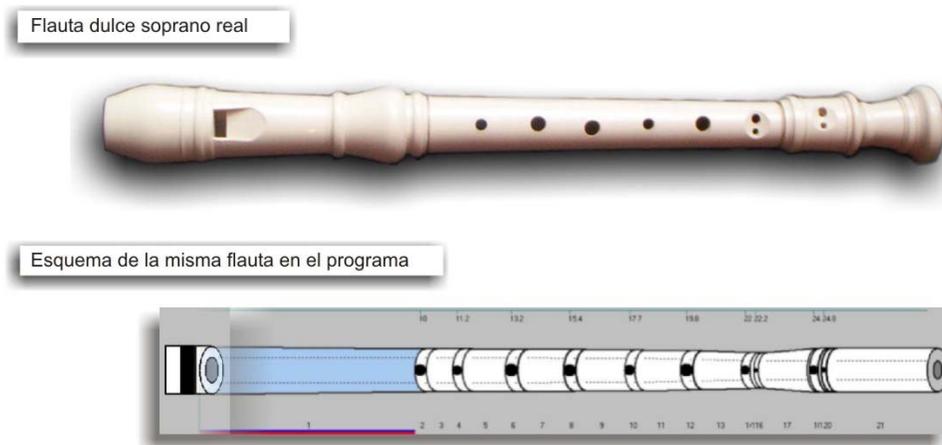
Para las pruebas, por su disponibilidad y economía, se escogió una flauta dulce soprano Yamaha de vaciado plástico. Se llevaron a cabo mediciones geométricas de la flauta, con objeto de poder separarla en elementos cónicos y de agujero, de acuerdo a las entradas de la interfaz del programa desarrollado. Las longitudes fueron medidas con ayuda de un vernier, y pueden observarse en las siguientes tablas.

Boquilla	
Anchura del canalillo [h]	$1.0 \pm 0.05$ mm
Longitud de la ventana [W]	$4.0 \pm 0.05$ mm
Anchura de la ventana [D]	$10.0 \pm 0.05$ mm

Resonador			
Número de agujero	Diámetro (mm)	Posición central medida desde la boquilla (mm)	Grosor (mm)
1	$5.50 \pm 0.005$	$100.00 \pm 0.005$	$4.00 \pm 0.005$
2	$4.50 \pm 0.005$	$112.00 \pm 0.005$	$4.00 \pm 0.005$
3	$6.00 \pm 0.005$	$132.00 \pm 0.005$	$4.00 \pm 0.005$
4	$5.70 \pm 0.005$	$154.00 \pm 0.005$	$4.00 \pm 0.005$
5	$4.90 \pm 0.005$	$177.00 \pm 0.005$	$4.00 \pm 0.005$
6	$5.90 \pm 0.005$	$198.00 \pm 0.005$	$4.00 \pm 0.005$
7	$4.30 \pm 0.005$	$220.00 \pm 0.005$	$4.00 \pm 0.005$
8	$2.25 \pm 0.005$	$222.00 \pm 0.005$	$4.00 \pm 0.005$
9	$4.40 \pm 0.005$	$246.00 \pm 0.005$	$5.95 \pm 0.005$
10	$2.50 \pm 0.005$	$248.00 \pm 0.005$	$5.95 \pm 0.005$
Final	----	295.00	----

Con estos datos, se calcularon las dimensiones de los elementos separados (secciones cónicas y de agujero). Estos elementos fueron introducidos al programa, de manera que la

flauta real fuera representada virtualmente dentro del programa. En la figura 4.1 puede verse una comparación entre la flauta real y el esquema gráfico en el programa.



*Figura 3.1: Comparación entre la flauta dulce soprano real y el esquema mostrado en la ventana de edición de instrumentos, una vez introducidos los datos medidos.*

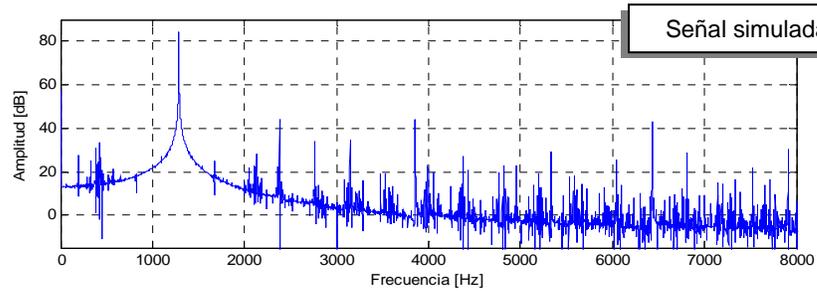
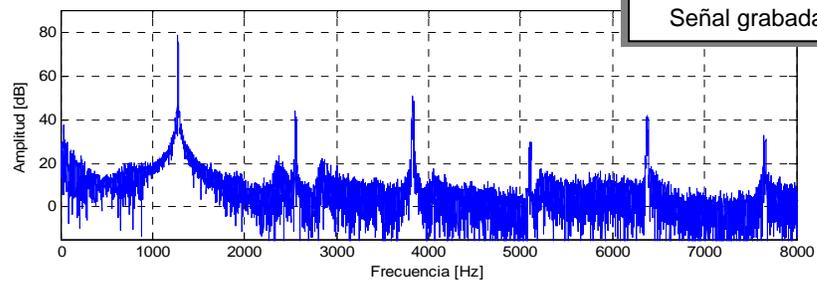
Posteriormente se realizaron las pruebas comparativas de sonido. Para ello, se hicieron grabaciones de la flauta soprano con distintas digitaciones (notas) y, paralelamente, se realizaron simulaciones en el programa con las digitaciones correspondientes. La nomenclatura de las notas se basa en la convención de La 4 (A4) correspondiente a 440 Hz.

Los registros sonoros son comparados tanto en su espectro de frecuencias, como en la forma de onda. Estas comparaciones pueden observarse de la figura 3.2 a la 3.7. En la presentación de los resultados, se tienen en cuenta las siguientes consideraciones:

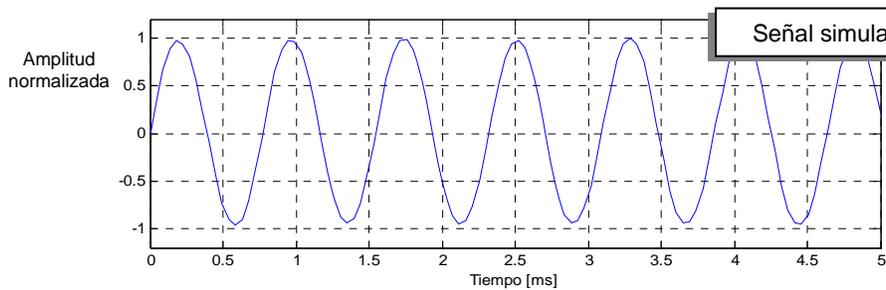
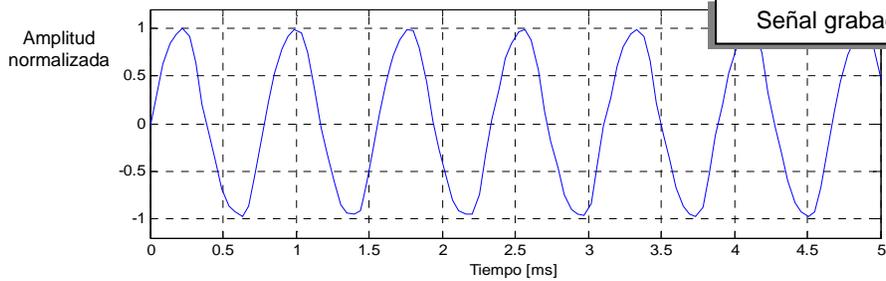
- Las amplitudes de las grabaciones y los sonidos simulados se encuentran normalizadas, para facilitar la comparación. De igual forma, las amplitudes de los espectros de frecuencia se miden en decibeles, cuyo valor de referencia es 1.
- En las simulaciones, la presión de entrada en la boquilla fue ajustada para corresponder a la interpretación más cercana al sonido grabado, y se conserva constante en toda la interpretación de la nota.
- El análisis de espectro corresponde a un tiempo de análisis de dos segundos.
- Todas las simulaciones se llevaron a cabo utilizando el modelo de excitación de McIntyre et al. El modelo de Verge mostró problemas graves de estabilidad numérica en su implementación, por lo que no fue posible realizar pruebas acústicas con tal modelo. La inestabilidad se origina en el segundo término de la ecuación (1.36), la cual, en pocos ciclos, diverge numéricamente.

## Nota Re# 6 (D#6)

### Espectro de Frecuencias



### Forma de onda



### Digitación

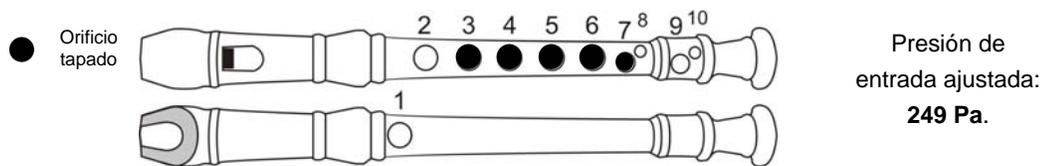
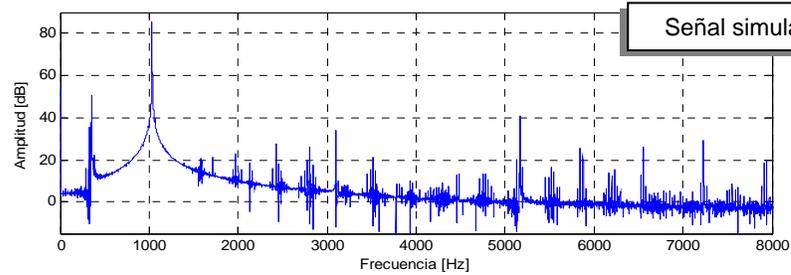
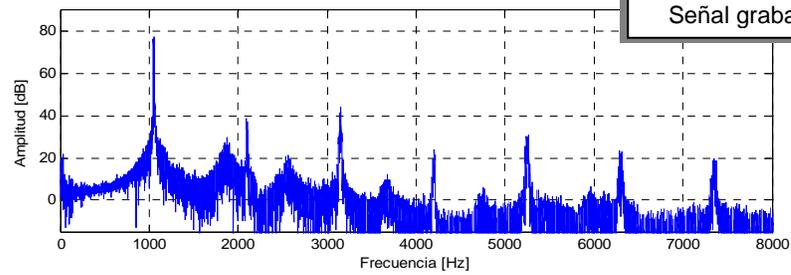


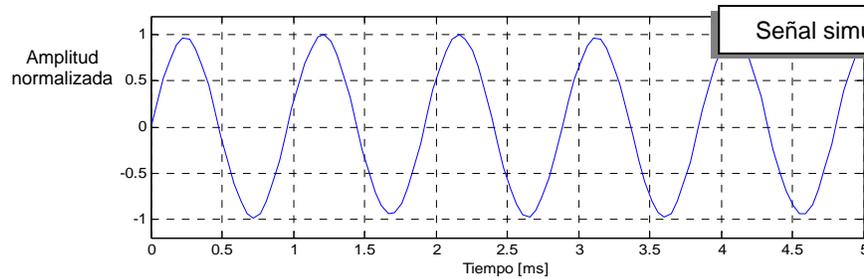
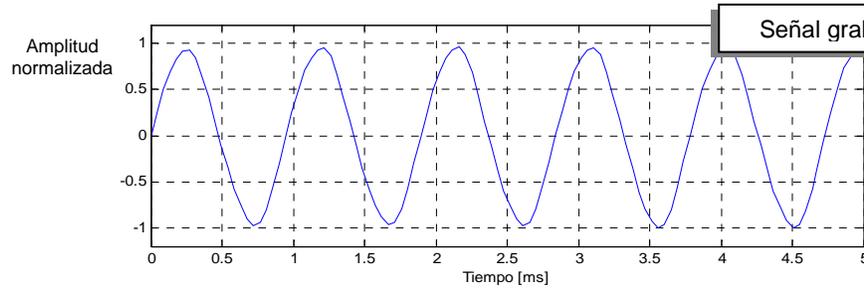
Figura 3.2: Comparación de señales acústicas para la digitación de la nota Re# 6.

## Nota Do 6 (C6)

### Espectro de Frecuencias



### Forma de onda



### Digitación

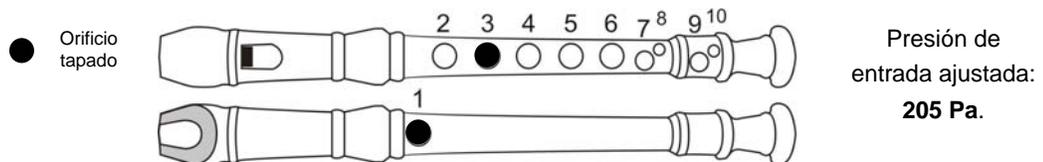
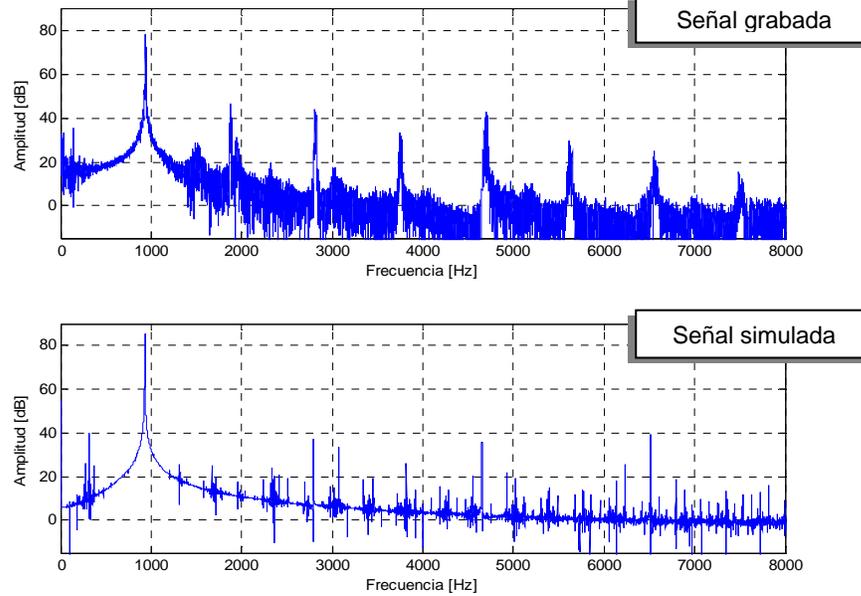


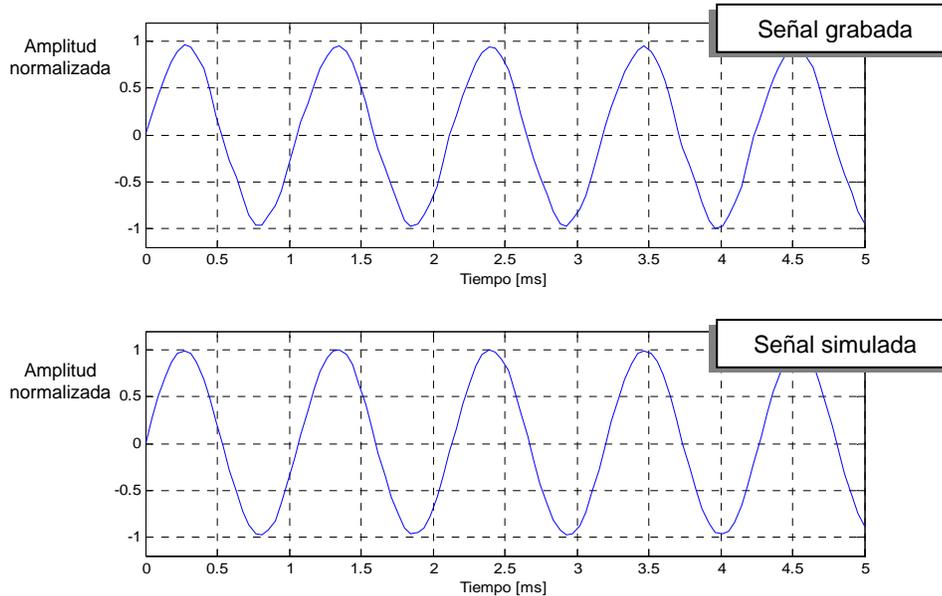
Figura 3.3: Comparación de señales acústicas para la digitación de la nota Do 6.

## Nota La# 5 (A#5)

### Espectro de Frecuencias



### Forma de onda



### Digitación

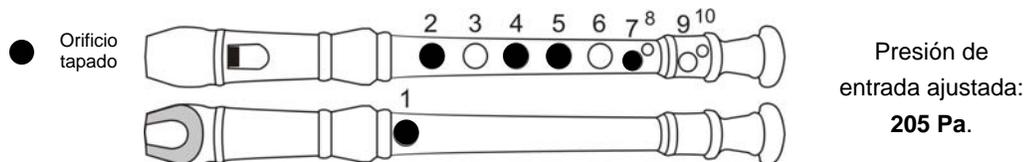
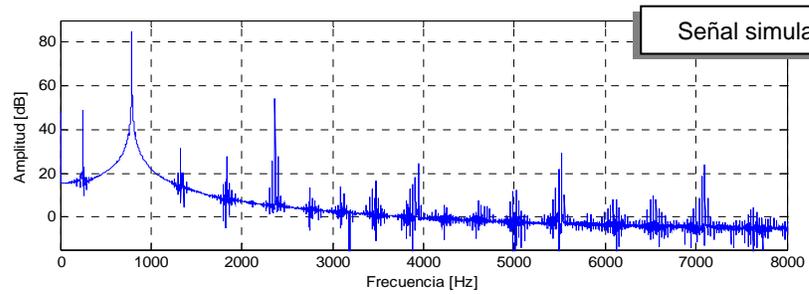
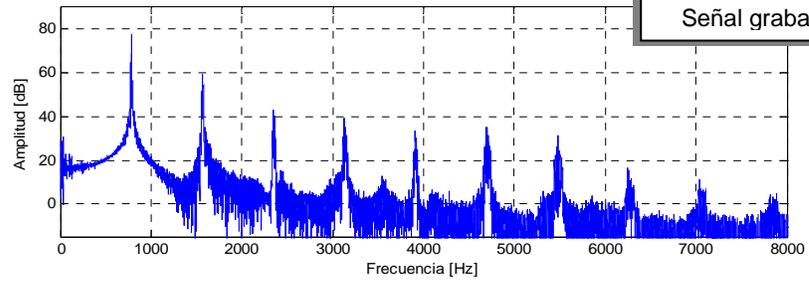


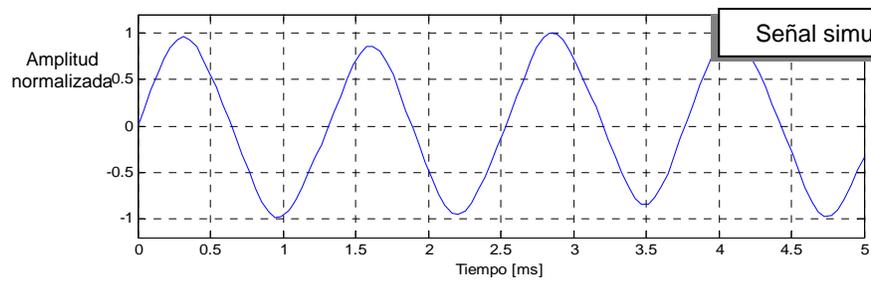
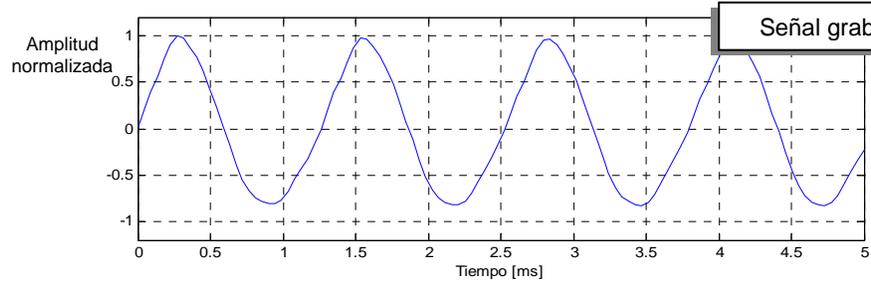
Figura 3.4: Comparación de señales acústicas para la digitación de la nota La sostenido 5.

## Nota Sol 5 (G5)

### Espectro de Frecuencias



### Forma de onda



### Digitación

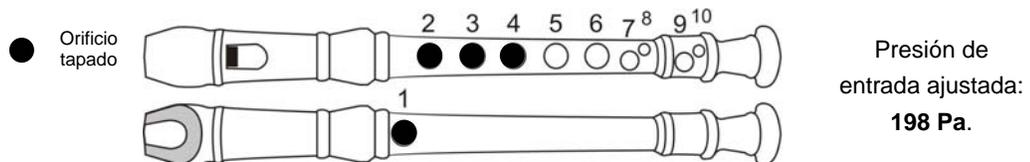
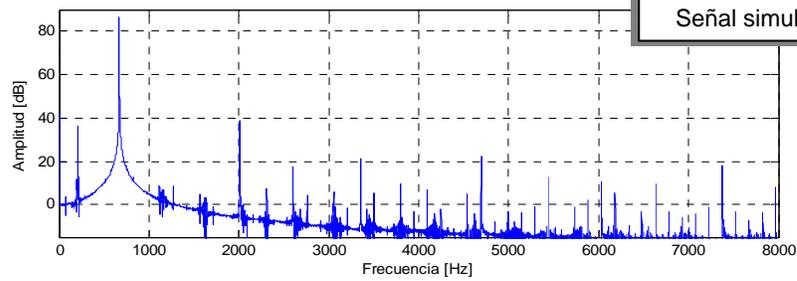
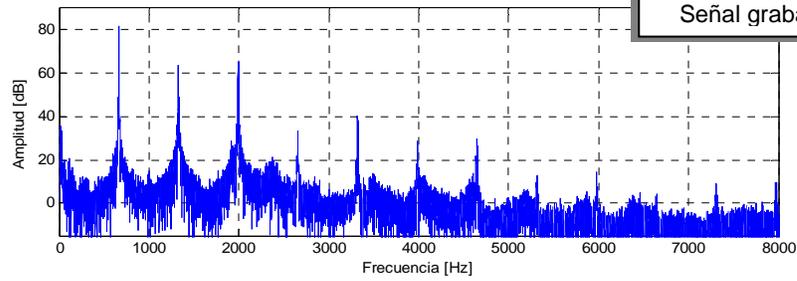


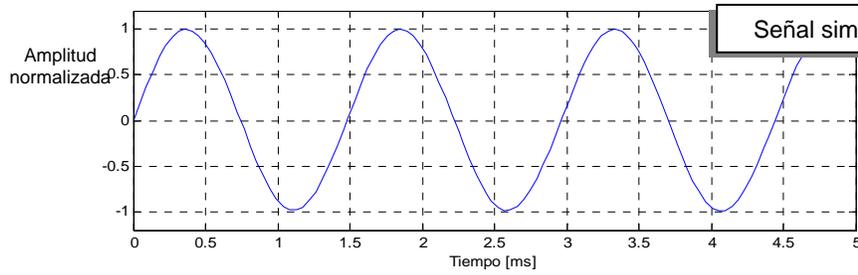
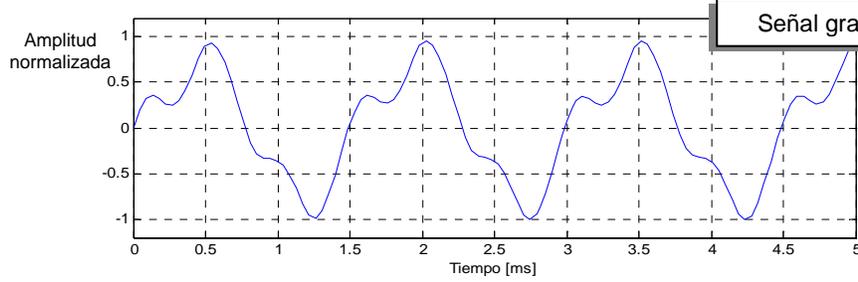
Figura 3.5: Comparación de señales acústicas para la digitación de la nota Sol 5.

## Nota Mi 5 (E5)

### Espectro de Frecuencias



### Forma de onda



### Digitación

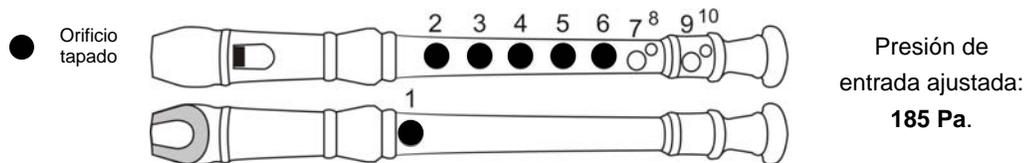
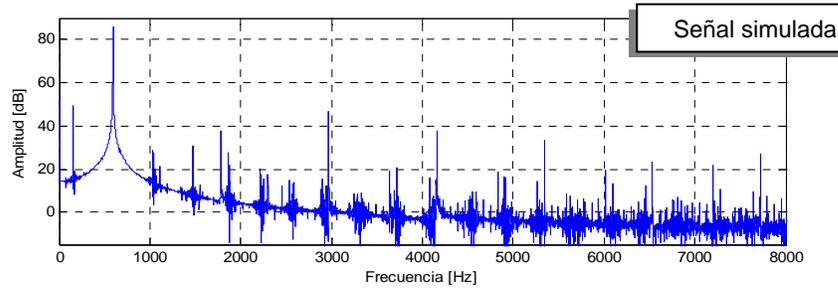
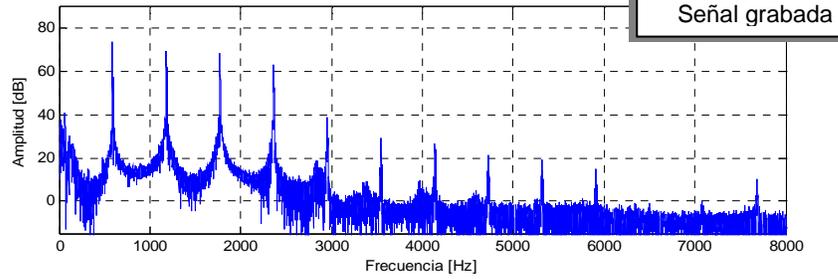


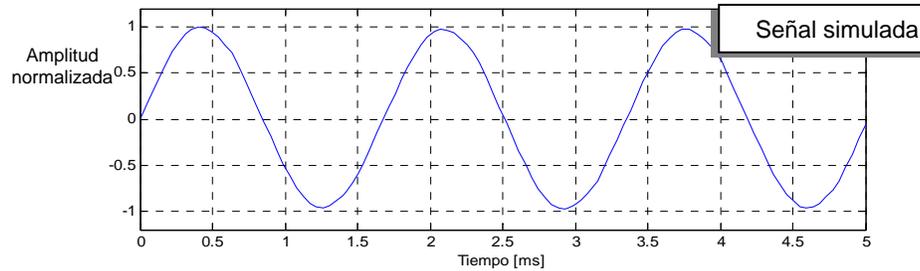
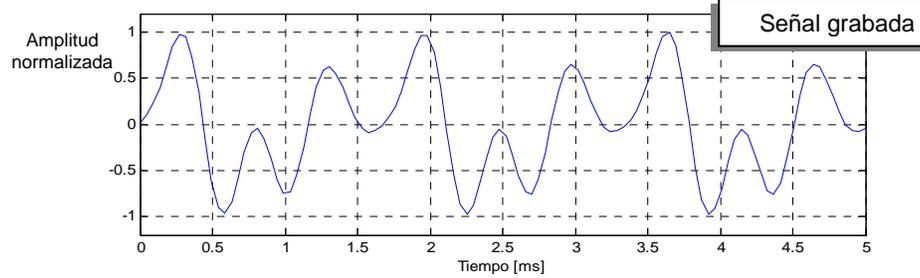
Figura 3.6: Comparación de señales acústicas para la digitación de la nota Mi 5.

## Nota Re 5 (D5)

### Espectro de Frecuencias



### Forma de onda



### Digitación

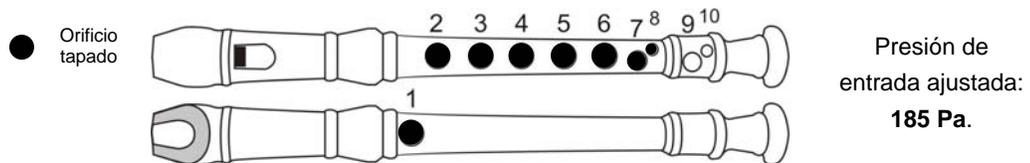


Figura 3.7: Comparación de señales acústicas para la digitación de la nota Re 5.

### **3.2 Discusión de resultados**

Los resultados de las pruebas comparativas arrojan información importante con respecto al desempeño del programa.

Como puede observarse en los espectros de frecuencia mostrados en las figuras 3.2 a 3.7, la correspondencia con la frecuencia fundamental es muy alta y se distingue en todas las gráficas. Los armónicos, a su vez, tienen variaciones.

Los armónicos, notables en todas las mediciones de la flauta real, son de amplitud menor en los espectros de las simulaciones. Esta discrepancia no es muy significativa en las mediciones de las notas Re# 6, Do 6, y Si 5, donde los armónicos de ambas señales permanecen a más de 35 dB por debajo de la frecuencia fundamental.

En la grabación de la flauta real, a partir de la nota Sol 5, los primeros armónicos se hacen mayores. El segundo armónico de la nota Sol 5 está 18 dB debajo de la fundamental. No ocurre así con la simulación, donde los armónicos permanecen a un nivel comparable con el de las notas más agudas.

En las notas más graves (Mi 5 y Re 5), la discrepancia aumenta visiblemente, tanto en el espectro, como en la forma de onda. Aún así, se conserva la correspondencia con la frecuencia fundamental.

Esta discrepancia en las comparaciones de las notas más graves se atribuye al modelo del excitador. El modelo de McIntyre es simple y tiene limitaciones. En la práctica, el modelo de la excitación no es capaz de provocar, con suficiente intensidad, las vibraciones acústicas en modos más elevados del resonador. Puesto que las notas más graves de la flauta dulce tienden a tener armónicos de mayor intensidad, las discrepancias aumentan en este tipo de notas.

Las presiones ajustadas para la simulación oscilaron entre los 185 Pa y los 249 Pa, los cuales son valores verosímiles. Como dato de referencia, se sabe que un flautista común, a baja intensidad, produce un flujo volumétrico aproximado de  $5 \times 10^{-4} \text{ m}^3/\text{s}$  [23], el cual corresponde a una presión de 151 Pa.



# *Conclusiones y recomendaciones*

## **CR.1 Conclusiones**

Finalmente, puede concluirse que el presente proyecto de tesis tuvo las siguientes aportaciones:

- Se llevó a cabo una investigación documental sobre el modelado matemático de instrumentos de viento tipo flauta. A partir de esta investigación, y principalmente de las propuestas de McIntyre [1] y Keefe [12], se programaron los algoritmos de simulación.
- Se diseñó una interfaz gráfica completa y detallada, con el fin de simplificar las tareas de entrada de datos (diseño del instrumento) y obtención de resultados (gráficas y sonidos simulados). Esta interfaz es la que permite a cualquier usuario diseñar su propio instrumento y realizar simulaciones, sin tener conocimientos concretos de física acústica o de computación.
- Los módulos operativos de los modelos de simulación fueron programados en bibliotecas separadas a la interfaz de ventanas, en lenguajes C y C++, para garantizar su portabilidad a través de distintas plataformas.
- Al momento de realizar simulaciones, el tiempo de ejecución de la aplicación fue del orden de uno a diez segundos. Estos intervalos son aceptables, si se toma en cuenta la gran cantidad de procesos iterativos contenidos en los algoritmos internos del programa.
- Se realizaron pruebas comparativas entre los espectros y las formas de onda producidos con una flauta soprano real y con una flauta idéntica simulada en el programa. En estas pruebas, las simulaciones tuvieron una coincidencia consistente con el tono fundamental de las grabaciones de la flauta real. La coincidencia de espectros entre el sonido simulado y el sonido grabado se acentúa en las notas más agudas, donde los armónicos del sonido grabado son de menor amplitud. En las notas más graves, la amplitud de los armónicos de la simulación se mantuvo muy por debajo (de 10 a 40 dB) de los armónicos de la señal grabada. Esta discrepancia se atribuye a limitaciones del modelo del excitador.

## ***CR.2 Recomendaciones para trabajos posteriores***

Algunas tareas que podrían ser de utilidad para proyectos posteriores sobre el presente trabajo de tesis son:

- Programar modelos más detallados para el mecanismo excitador, los cuales podrían mejorar el desempeño de las simulaciones en los armónicos. En este rubro, es conveniente estudiar con más detenimiento el modelo de Verge, para poder implementarlo exitosamente en el programa.
- Añadir nuevos modelos de excitador, los cuales permitirían simular otro tipo instrumentos de viento, como los de boquilla tipo caña. La estructura modular del programa facilita la inserción de nuevos modelos.
- Implementar optimizaciones en el código operativo, como el algoritmo de convolución rápida, lo cual reduciría el tiempo de ejecución del programa.
- Realizar pruebas de la interfaz con usuarios, con objeto de obtener retroalimentación y mejorar el entorno de trabajo del programa.
- Realizar pruebas acústicas con usuarios, con el fin de determinar cualitativamente el desempeño del programa.
- Programar interfaces gráficas que también puedan ser portadas entre distintas plataformas.

# Bibliografía

- [1] McIntyre, M. E., Schumacher, R. T., Woodhouse, J.  
"On the oscillations of musical instruments". J. Acoust. Soc. Am., Vol. 74, No. 5, pp. 1325-1345 (1983).
- [2] Suárez Hernández, María Angélica.  
*Medición de funciones de reflexión acústica en instrumentos musicales de viento*. Tesis de Licenciatura, Facultad de Ingeniería UNAM, 2004.  
Tutor: Dr. Felipe Orduña Bustamante.
- [3] Orduña Bustamante, Felipe., Elizalde Douce, Roberto.  
"Desarrollo de un simulador de instrumentos musicales de viento". En Memorias SOMI XIV Congreso de instrumentación, Tonanzintla, Puebla, México, 1999.
- [4] Fletcher, Neville H., Rossing, Thomas D.  
*The Physics of Musical Instruments*, Segunda Edición. Springer-Verlag, EE UU, 1998.
- [5] Scavone, G. P., Lakatos, S.  
"Recent developments in woodwind instrument physical modeling." 17th International Congress on Acoustics, Rome, Italy. Stanford University, 2001.
- [6] Kinsler, Lawrence E., et al.  
*Fundamentals of acoustics*. Tercera edición. John Wiley & Sons, EE UU, 1982.
- [7] Verge, Marc-Pierre., Fabre, Benoit., Mahu, W. E. A., Hirschberg, A., Van Hassel, R. R., Wijnands, A. P. J., De Vries J. J., Hogendoorn, C. J.  
"Jet formation and jet velocity fluctuations in a flue organ pipe". J. Acoust. Soc. Am., Vol. 95, No. 2, pp. 1119-1132 (1994).
- [8] Berg, Richard E., Stork, David G.  
*The Physics of Sound*. Prentice Hall, EE UU, 1982.
- [9] Verge, Marc-Pierre., Hirschberg, A., Caussé, R.  
"Sound production in recorderlike instruments II. A simulation model". J. Acoust. Soc. Am., Vol. 101, No. 5, Pt.1, pp. 2925-2939 (1997).

- [10] Bartsch, Mark A.  
“A Hybrid Waveguide Model of the Transverse Flute”.  
[http://home.earthlink.net/~bartscma/hybrid\\_waveguide\\_model.pdf](http://home.earthlink.net/~bartscma/hybrid_waveguide_model.pdf)  
University of Dayton, Department of Electrical and Computer Engineering (2005).
- [11] Verge, Marc-Pierre., Fabre, Benoit., Hirschberg, A., Wijnands, A. P. J.  
“Sound production in recorderlike instruments I. Dimensionless amplitude of the internal acoustic field”. J. Acoust. Soc. Am., Vol. 101, No. 5, Pt.1, pp. 2914-2924 (1997).
- [12] Keefe, Douglas H.  
“Woodwind air column models”, J. Acoust. Soc. Am., Vol. 88, No. 1, pp. 35-51 (1990).
- [13] Plitnik, George R., Strong, William J.  
“Numerical method for calculating input impedances of the oboe”, J. Acoust. Soc. Am., Vol. 63, No. 3, pp. 816-825 (1979).
- [14] Brown, Adrian.  
*The Recorder: A Basic Workshop Manual*. Dolce Edition, Gran Bretaña, 1989.
- [15] Kientzle, Tim.  
*A Programmer's Guide to Sound*. Addison Wesley, EE UU, 1988.
- [16] Oppenheim, Alan V., Willsky, Alan S.  
*Signals & Systems*. Segunda edición. Prentice Hall, EE UU, 1997.
- [17] Reisdorph, Kent.  
*Aprendiendo Borland C++ Builder 3 en 21 días*. Prentice Hall, México, 1999.
- [18] Kernighan, Brian W., Pike, Rob.  
*The Practice of Programming*. Addison Wesley, EE UU, 1999.
- [19] Ritchie, Dennis M.  
*The C Programming Language*. Segunda edición. Prentice Hall, EE UU, 1988.
- [20] Berryhill, John R.  
*C++ Scientific Programming*. John Wiley & Sons, EE UU, 2001.
- [21] Press, William H., Teukolsky, Saul A., Vetterling, William T., Flannery, Brian P.  
*Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Inglaterra, 1992.

[22] Kreyszig, Erwin.

*Matemáticas Avanzadas para Ingeniería: Volumen II.* Limusa, México, 1992.

[23] Martin, John.

*The Acoustics of the Recorder.* Moeck Verlag Celle, Alemania, 1994.