POLITECNICO DI MILANO

Modeling of Musical Instruments

Mirco Pezzoli
Riccardo R. De Lucia

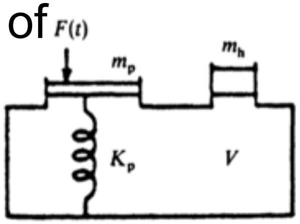
ISPL
Image and Sound Processing Lab

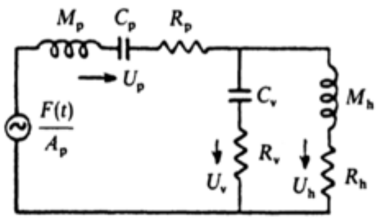
Modeling a dynamic system

- Given a real system, we want to study it through a **mathematical approach**.
- For **dynamic systems**, this translates into building a **set of differential equations** which can describe **relationships among values** in systems.
- The **model complexity** determines the **accuracy** of the system representation. E.g: an ideal string model cannot represent real strings inharmonicities.
- One **same system** can often be described through **different models**, some of which can be **equivalent**, and therefore **interchangeable**.

Physical system

Dynamic system = set of differential equations





Equivalent models

Requirements for a good sounding model

- Models are built to study system behaviors.
- Sound models are built to reproduce sounds from real instruments.
- **Tradeoff** between model **accuracy** and **computational effort** (real-time issues).
- Sound models deal with:
  - **Timbre quality**: how much the timbre for a synthesized instruments equals the timbre complexity of musical instruments. Timbre mostly deals with the time-evolution of sound (envelope).
  - **Timbre dynamic range**: how the synthesized instrument can be able to adapt to the playing intensity, by reproducing related timbrical variations accordingly.
  - **Playability**: how much is easy to control the synthetic instrument sound by the player. A good synthetic instrument should change its sound according to same principles and parameters as real instruments.

Characteristics of real instruments

- Very complex systems
- Their **sound behavior changes** according to the **playing style** and lots of other factors
- The sound highly depends on **the instrument system state**. The combination of a particular state and time/type of input can greatly modify the sound
- **Non linear interactions** (especially for stimulations) are ubiquitous (e.g: bow string interaction in violins)

Synthetic algorithms and Samplers

- **Synthetic algorithms**:
  - Goal: try to **reproduce the sound** instead of the system
  - Not linked to the physics of real instruments
  - Less computational complexity
  - Limited approach for real instruments: often they are not able to accurately reproduce the sound nuances and evolution in time
  - Real common types of synthesis: **additive**, **subtractive**, FM, granular

Real instrument sound synthesis is not a trivial task

- One first approach: **Samplers**
  - They reproduce huge collections of accurately recorded real instruments
  - Samplers sound as real instruments
  - Great samplers can reproduce different instruments playing styles by responding to input parameters (e.g. velocity)
  - Samplers require a lot of memory for storing all the samples

Physical modeling

- Goal: try to reproduce the **system behavior**, in terms of **physics**
- This approach automatically solves (or **attenuate**) problems such as **timbre quality, timbre dynamic range** and **playability**
- Physical modeling requires to start from the dynamic **differential equations** which describe the system
- Often, such equations are **not easy to be found**, and can become highly **complicated**
- This approach requires **great computational complexity**. That is why it has not been seriously taken into account until highly performing computers have become available.

$$\frac{dy}{dx} - x^2 + 3x - 2 = 0$$

$$\frac{dy}{dx} = x^2 - 3x + 2$$

$$dy = (x^2 - 3x + 2)dx$$

$$y = \int (x^2 - 3x + 2)dx$$

$$y = \frac{x^3}{3} - \frac{3x^2}{2} + 2x + C$$

Physical modeling starts from differential equations in order to represent musical instruments

Physical modeling techniques

- Starting from constitutive equations, several techniques have been developed for the implementation of sound synthesis algorithms
- Of course, we deal with **discrete algorithms**. A discretization process is always concerned
- **Discretization schemes** can give rise to **problems** (aliasing, accuracy, stability, etc)
- We can basically divide between:
    - Methods which focus on the **system description** (i.e. differential equations) (e.g. **time-stepping methods**)
    - Methods which focus on the **system solution** (e.g. **modal synthesis**, **digital wave guides**)

Time-stepping methods

- The PDE (partial differential equations) are **discretized in time and space**
- Partial differential operators are **approximated** through **finite differences**
- Finite differences equations describe the **next time instant value** for each discretized spatial point of the system
- For each temporal/spatial point, the next value depends only on **neighbouring points**

1) Being an approximation, finite differences equations can be obtained also from differential equations which cannot be solved analitically

2) Locality principles: separated systems can be connected by focusing only on their contact points.

Because of 1 and 2, non linearities can be ’easily’ inserted in the systems description

Finite differences

- Discretization schemes
- Each scheme has different properties in terms of **accuracy**, **stability**, etc
- Basic idea: the derivative is **approximated** through **a finite difference**
- For time-dependent functions, this allows to separate the **current value from the next instant value**: no need for analytical solutions
- Start from **Taylor expansion** for functions

$$f(a+h) = f(a) + f'(a)h + R_1(x)$$

$$\Downarrow$$

$$f'(a) \approx \frac{f(a+h) - f(a)}{h}$$

Solving the stiff and lossy string through FDM

Differential equation

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2} - \kappa^2 \frac{\partial^4 y}{\partial x^4}$$

Finite difference equation

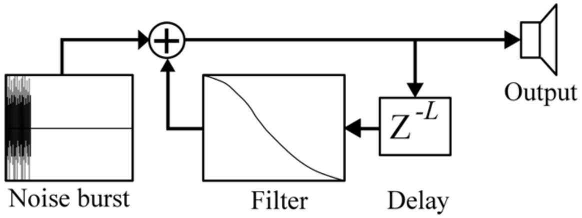
$$y_m^{n+1} = a_1(y_{m+2}^n + y_{m-2}^n) + a_2(y_{m+1}^n + y_{m-1}^n) + a_3 y_m^n + a_4 y_m^{n-1} + a_5(y_{m+1}^{n-1} + y_{m-1}^{n-1})$$

Boundary and initial conditions still need to be applied to complete the finite difference equation definition

$$\left.\frac{\partial^2 y}{\partial x^2}\right|_{x_{m_0},t_n} = \left.\frac{\partial^2 y}{\partial x^2}\right|_{x_M,t_n} = 0$$

Digital waveguide methods

- Suitable for models characterized by the **propagation of waves**. Therefore often used for **pipes and string models**
- They start from a **discretization of the general analytical solution** in the time domain (travelling wave solution)
- They have been developed starting from a more ancient technique: the **Karplus-Strong algorithm**



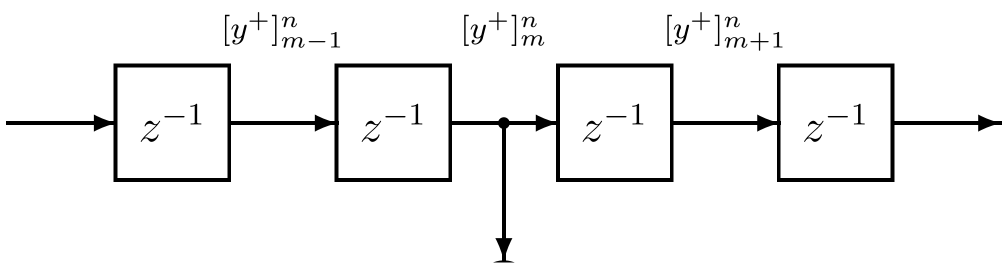
- Main idea: the signal is modeled according to the **D'Alembert solution**
- Therefore, we consider two waves travelling in opposite directions

$$y(x,t) = f(t - x/c) + h(t + x/c)$$

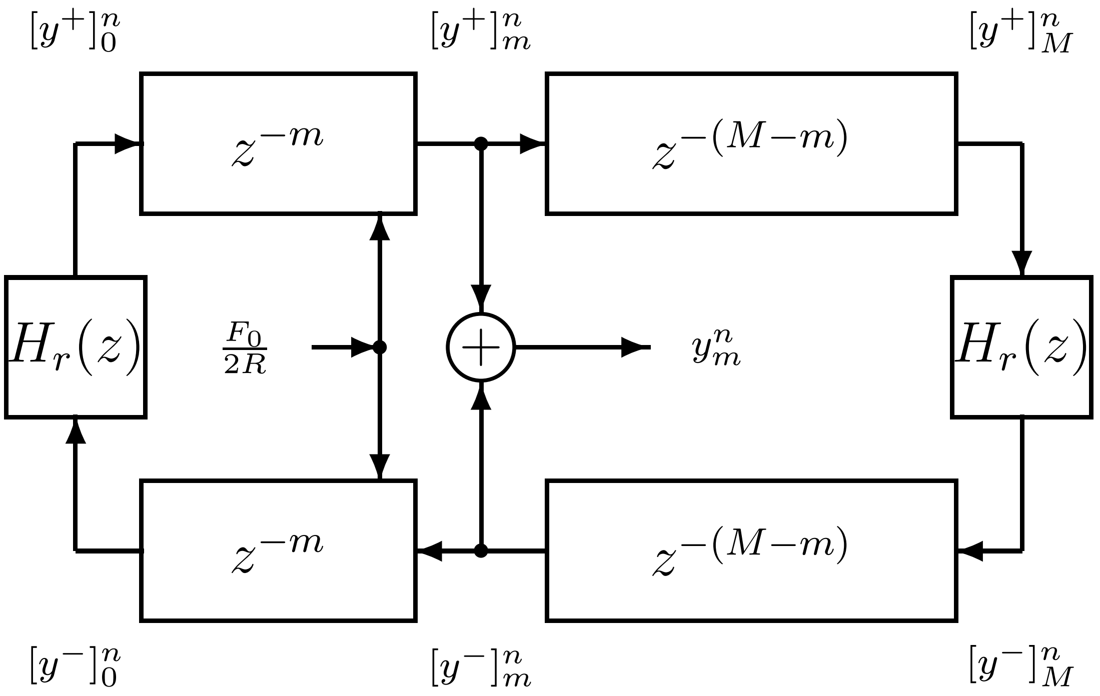
$$[y^+]_m^n = f(t_n - x_m/c) = f[(n-m)T] = f(n-m)$$
$$[y^-]_m^n = h(t_n + x_m/c) = h[(n+m)T] = h(n+m)$$

Digital waveguide: the delay line

- Core structure at the basis of digital waveguides
- It simulates a **M sample delay** for a **discrete signal** entering the delay line.
- The delay line can simulate the **propagation of a wave in a guide** (e.g a string, a tube).
- Hypothesis: the signal **does not undergo modifications while in the delay line**
- Dispersion, damping and other phenomena can be added through LTI filters **in one point of the delay line**
- The final delay guide becomes a generic **delay** connected to a **lumped system** which performs filtering



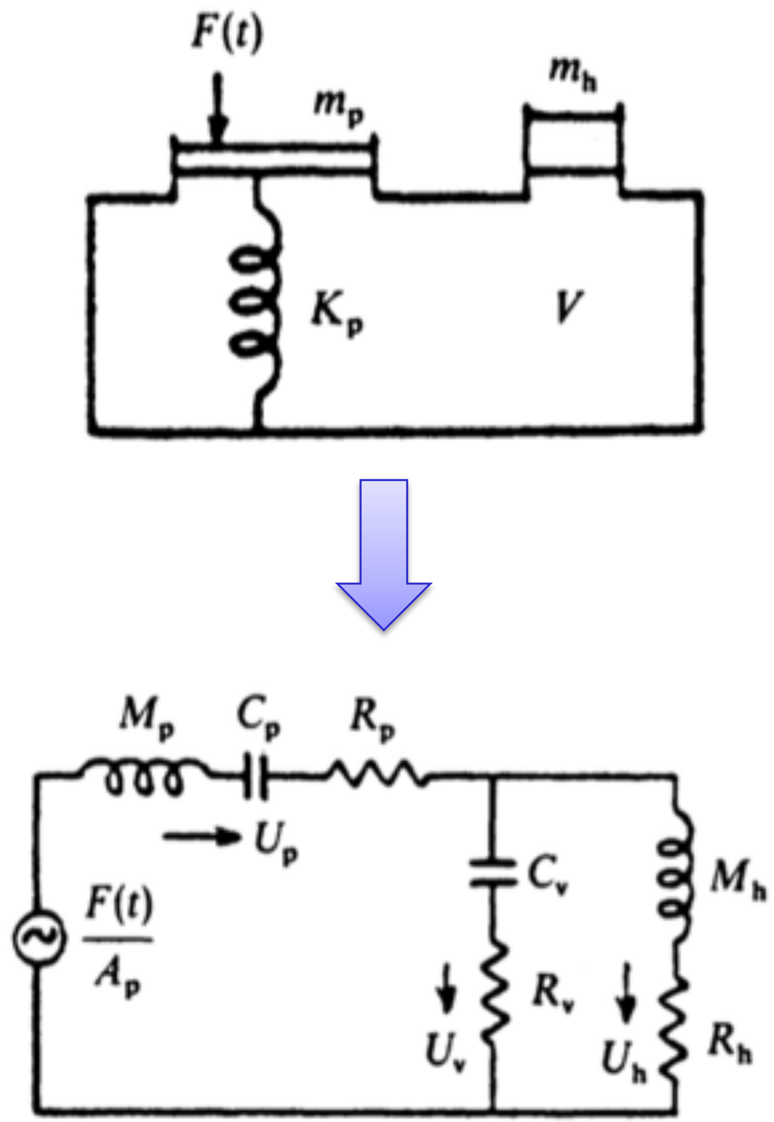
Digital waveguide scheme



Electric models for the guitar

One first simple model

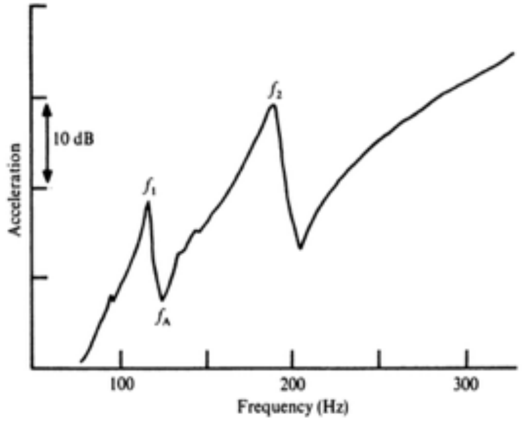
- We model a guitar with **rigid (non vibrating) back plate and ribs**.
- This simple model will not be representative of the complete behavior of a guitar.
- The **vibrational elements** are the **top plate** and the **cavity**.
- We model those 2 elements as 2 **lumped oscillators**.
- We then compute **an electrical equivalent circuit,** which can be analized with classical electric circuit theory.



One first simple model (cont'd)

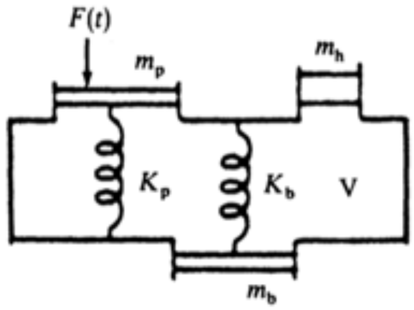
- This electrical system is described by a set of **two differential equations**. This means it is a **coupled system** with **two resonances**
- Because the excitation happens in the same place of the measure (the generator circuit branch, i.e the guitar bridge) we are dealing with a **driving point frf**
- Driving point admittances with more than one resonance are always characterized to have **one antiresonance between each couple of resonances**
- The **current** is associated with the **bridge velocity**, which is in turn associated with **sound** radiated from the instrument, thanks to the **Euler's equation**:

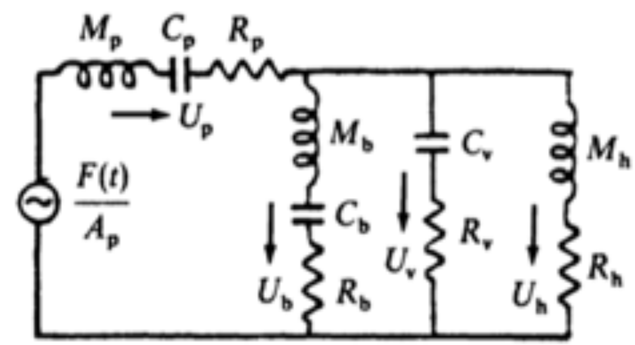
$$\rho_0 \frac{\partial v}{\partial t} = -\nabla p$$

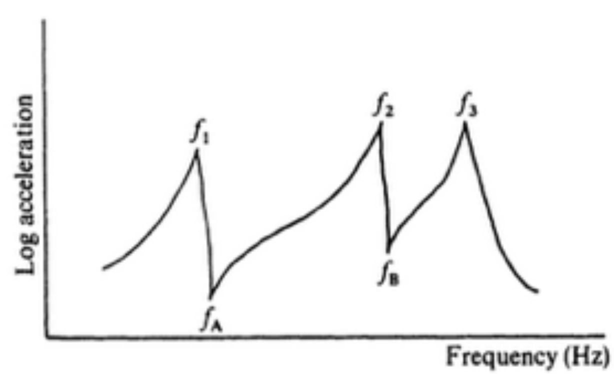


Refining the model (1° step)

- We add the **back plate vibration**, while keeping the ribs fixed
- The back plate couples to the top plate through the enclosed air
- By adding a new spring and mass, the system now has **three coupled simple oscillators**
- We therefore predict **three resonances** in the corresponding frf





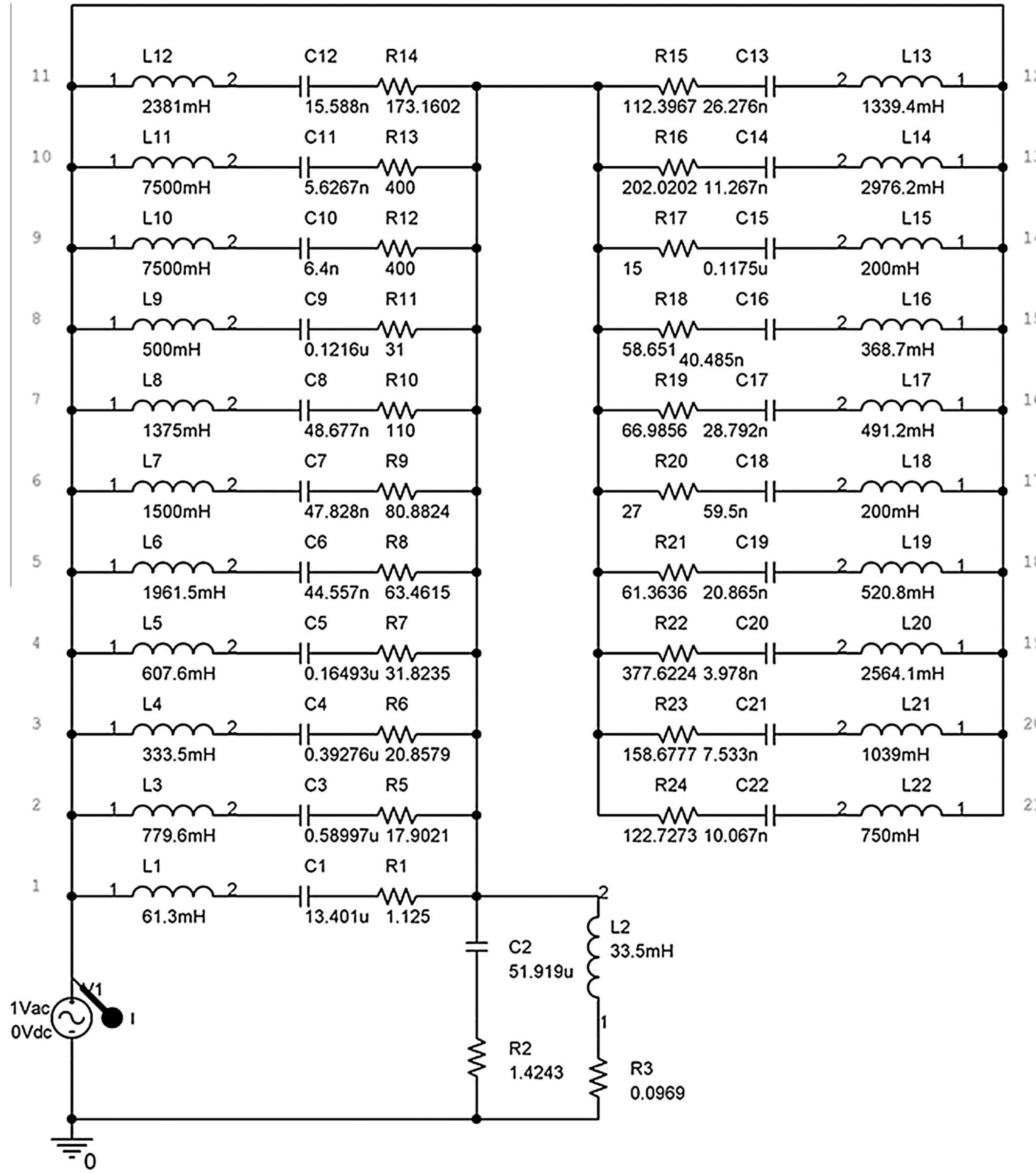


Refining the model (2° step)

- A real guitar top plate has more than one **resonances**, which we need to add to the model
- **For this case we will consider again the system with fixed back plate**.
- Resonances are modeled by inserting a **filter bank** in the circuit. Each **RLC branch** will model a particular filter with one resonance.
- RLC branches values are empirically **tuned** to obtain back the resonances **measured from a real guitar**.
- We are interested in studying the **time response** of the guitar according to a **pluck** solicitation. We therefore add a particular **signal generator circuit** which can emulate the **plucked string vibration**.
- We consider an **ideal string** with no damping or stiffness. The real string closest to this representation is the **E1 string for guitars**. Therefore we will study the time response of the guitar when the E1 string is plucked.

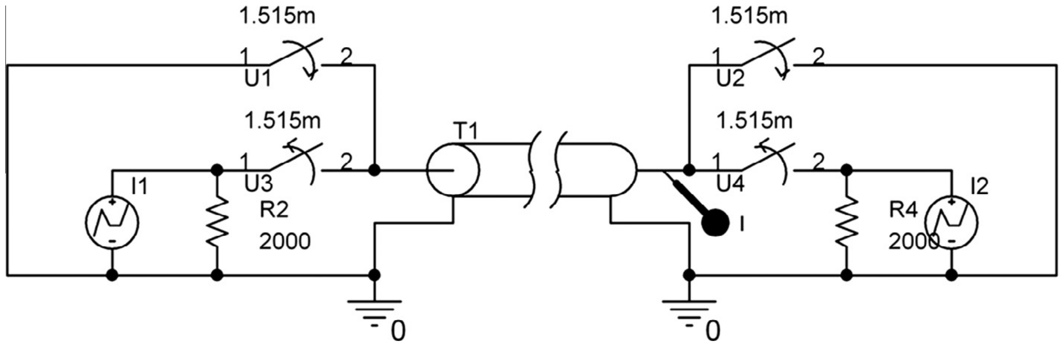
The extended model

- 20 **top plate resonances** are added.
- The **current** probe is placed where the current represents the **bridge velocity** (i.e guitar sound).
- **Resistors** in the circuit are intended as **damping sources** attenuating the input signal.



Pluck signal generator circuit

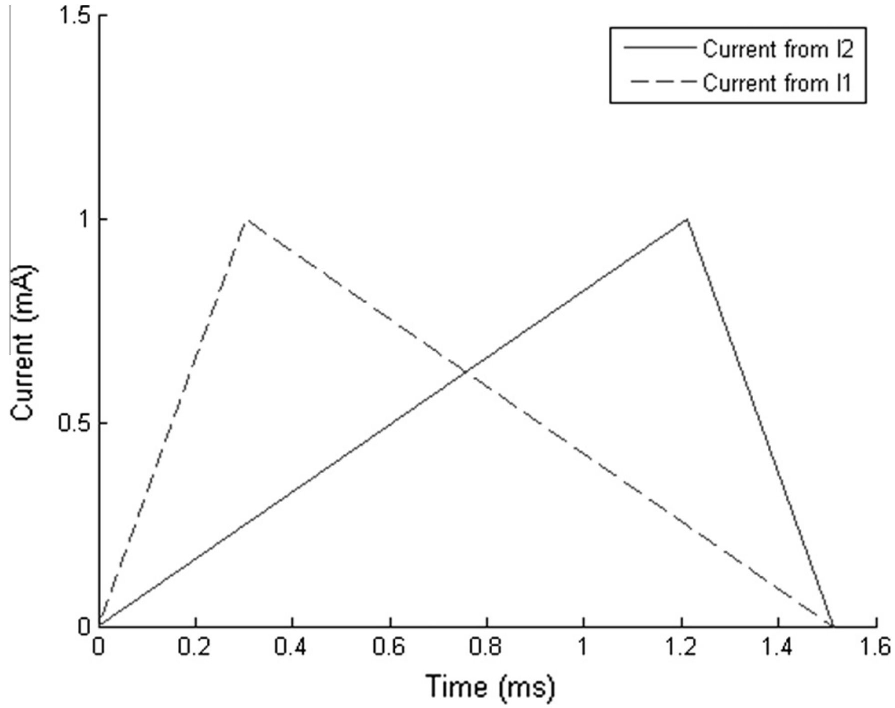
- **Standing** waves in the string are modeled through a transmission line, which allows to take into account **wave reflections** at the string boundaries
- The **string vertical displacement** at each spatial point is related to the corresponding **voltage** at the same place in the transmission line



- The transmission line is fed on both sides with two **symmetrical triangular current pulses**. Their **duration** is linked to **half of the string fundamental frequency** of vibration
- The **shape of the pulses** determines the initial **shape of the string** plucked at a certain point on its length

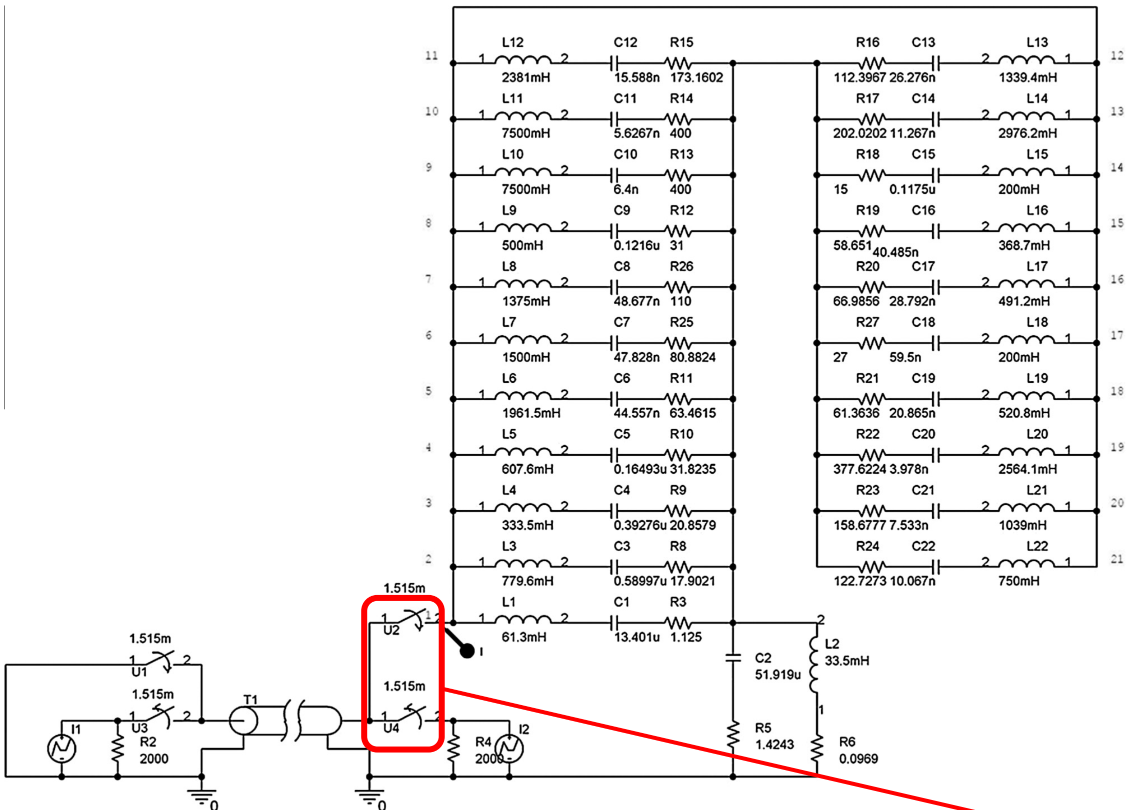
Pluck signal generator circuit (cont'd)

- These pulses correspond to two feasible propagating **D'Alembert solutions** for the transverse wave propagation on the string

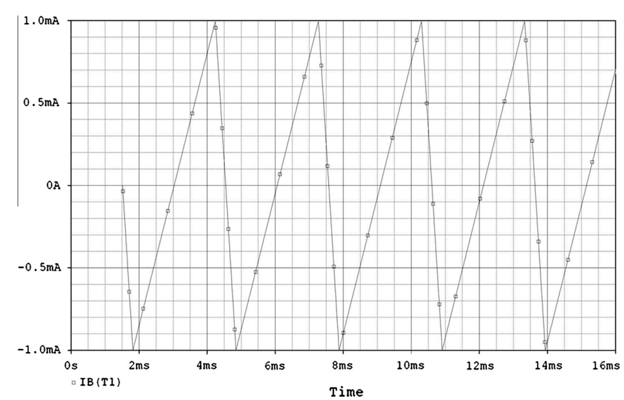


Pulse duration = 1.515 ms corresponds to f=329.5 Hz for the string (E1 guitar string fundamental frequency)

The final circuit



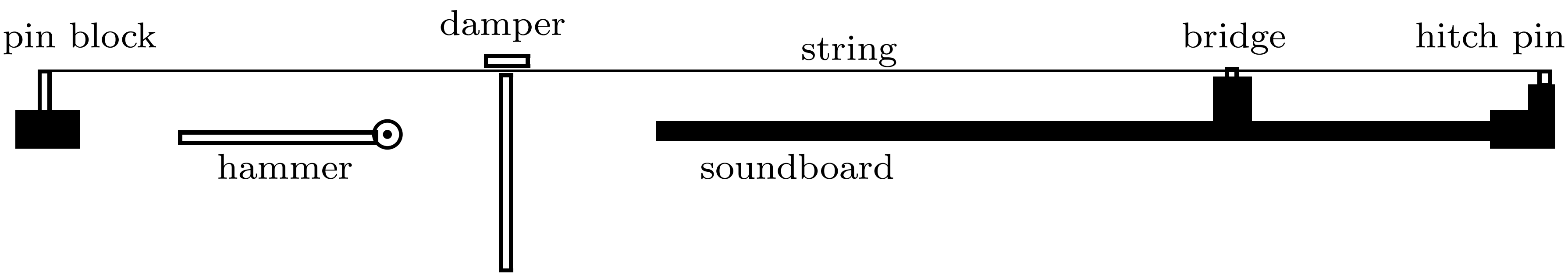
After the current sources are switched off, the right end of the transmission line feeds the circuit with a triangular waveform that simulates the solicitation coming from a plucked string on a guitar body.



String-hammer interaction within a piano

Simplified interaction model

- The **hammer motion mechanism** (capstan, whippen, jack, knuckle, repetition lever) is **omitted**
- Pianos have three string for each note (tricord). Here we consider **only one string**
- We omit the **damper** and the coupling with the **soundboard**
- Therefore we finally focus on a **real string** (damping+stiffness) left pinned and interacting with the **bridge** at the right, impacted through a **piano hammer**



Piano hammer

- **Hard wood core** covered with two layers of **wool felt** of varying thickness that increases from treble to bass.
- Dynamic hardness properties: **the hammer hardness varies with hammer velocity** according to the material properties of the felt. **Higher velocity** causes the felt compression to **harden the hammer**, thus soliciting higher harmonics.
- This process is highly non-linear (non-linear hardening)
- The **dynamics of playing**, through the hammer, influences the **timbre** of the played note.

Piano hammer (cont'd)

- The hammer felt can be considered as a **nonlinear spring**, whose stiffness increases with compression
- The complete hammer model is a **lumped mass attached to a nonlinear spring**
- We have finally a non-linear force interaction with the string, which is given by the following power law:

$$F_H = K\xi^p$$

Stiffness exponent: it describes how the stiffness changes with the force

$\xi(t)$ *is a time dependent function which describes the felt compression*

Hammer stiffness

Hammer force expression

- The interaction is modeled through a time dependent force exerted by the hammer **on a specific point $\mathbf{x_0}$**, and is expressed by:

$$M_H \frac{d^2\eta}{dt^2} = -F_H(t) - b_H \frac{d\eta}{dt}$$

Air damping

Hammer mass

η*(t) is the hammer displacement wrt the string equilibrium position*

- The hammer force can be furtherly refined as:

$$F_H(t) = \phi[\xi(t)] = K\xi^p, \qquad \xi(t) = |\,\eta(t) - y(x_0, t)|$$

Hammer position

String vertical in $x = x_0$ position at instant t

Refined model

- The refined model takes into account a **stiff and lossy string** with **scalar impedance on both ends**, excited by the hammer, and implements the **hammer string interaction** previously introduced.
- The relative differential equation **has no analytical solution**.
- We therefore solve the model through an **approximation**, taking advantage of the finite **finite difference method**.

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2} - \kappa^2 \frac{\partial^4 y}{\partial x^4} - 2b_1 \frac{\partial y}{\partial t^2} + 2b_2 \frac{\partial^3 y}{\partial x^2 \partial t} + \boxed{\rho^{-1} f(x, x_0, t)}$$

Hammer mass density

$$f(x, x_0, t) = f_H(t) g(x, x_0)$$

$$f_H(t) = F_H(t) \left(\int_{x_0 - \delta x}^{x_0 + \delta x} g(x, x_0) dx \right)^{-1}$$

g is a dimensionless spatial window of length 2δ. This window limits in space the force exerted by the hammer on the string

Boundary conditions

- Boundary conditions at the string right end (at the bridge). x_M is the coordinate at the bridge end.

$$\frac{\partial^3 y}{\partial x^3} = -\frac{c^2}{\kappa^2}\frac{\partial y}{\partial x} + \frac{\zeta_b c}{\kappa^2}\frac{\partial y}{\partial t} \qquad \zeta_b = \frac{R_b}{\rho c} \qquad \left.\frac{\partial^2 y}{\partial x^2}\right|_{x_M, t_n} = 0$$

- At the bridge the string actually cannot move. It only acts with a force F_b on the bridge:

$$F_b = R_b \frac{\partial y}{\partial t} \quad F_b = -T_e \frac{\partial y}{\partial x} + \kappa^2 \rho \frac{\partial^3 y}{\partial x^3} \qquad -T_e \frac{\partial y}{\partial x} + \kappa^2 \rho \frac{\partial^3 y}{\partial x^3} = R_b \frac{\partial y}{\partial t}$$

- Boundary conditions at the string left hinged end. x_M is the coordinate at the bridge end. x_0 here stands for the string coordinate at its left end (not the interaction point with the hammer)

$$\frac{\partial^3 y}{\partial x^3} = -\frac{c^2}{\kappa^2}\frac{\partial y}{\partial x} + \frac{\zeta_l c}{\kappa^2}\frac{\partial y}{\partial t} \qquad \zeta_l = \frac{R_l}{\rho c} \qquad \left.\frac{\partial^2 y}{\partial x^2}\right|_{x_0, t_n} = 0$$

- Again, the string cannot move at the left hinged edge.

Corresponding finite difference equation

- The wave equation is approximated by the following finite difference equation:

$$y_m^{n+1} = a_1(y_{m+2}^n + y_{m-2}^n) + a_2(y_{m+1}^n + y_{m-1}^n)$$
$$+a_3 y_m^n + a_4 y_m^{n-1} + a_5(y_{m+1}^{n-1} + y_{m-1}^{n-1}) + \mathrm{a_F F_m^n}$$

$$F_m^n = F_H(n) g(m, m_0)$$

- **m** is the **spatial index** (ranging from 0 to M), while **n** is the **time index** (ranging from 0 to an arbitrary value)
- This equation is **valid only in the interior of the string**. There is no m-2 for m=0 and m+2 for m=M
- **Boundary conditions** need to be imposed as in the differential equation case for such points
- In this case we need to set **two conditions for the first and last two discrete points** at the string edges

$$a_1 = \frac{-\lambda^2 \mu}{1 + b_1 T}$$

$$a_2 = \frac{\lambda^2 + 4\lambda^2 \mu + \nu}{1 + b_1 T}$$

$$a_3 = \frac{2 - 2\lambda^2 - 6\lambda^2 \mu - 2\nu}{1 + b_1 T}$$

$$a_4 = \frac{-1 + b_1 T + 2\nu}{1 + b_1 T}$$

$$a_5 = \frac{-\nu}{1 + b_1 T}$$

Finite difference modeling parameters

$$a_1 = \frac{-\lambda^2\mu}{1 + b_1 T}$$

$$a_2 = \frac{\lambda^2 + 4\lambda^2\mu + \nu}{1 + b_1 T}$$

$$a_3 = \frac{2 - 2\lambda^2 - 6\lambda^2\mu - 2\nu}{1 + b_1 T}$$

$$a_4 = \frac{-1 + b_1 T + 2\nu}{1 + b_1 T}$$

$$a_5 = \frac{-\nu}{1 + b_1 T}$$

$$\mu = \kappa^2 / c^2 X^2$$

$$\nu = 2 b_2 T / X^2$$

T = *temporal resolution (time step)*

X = *spatial resolution (spatial step)*

κ = *string stiffness coefficient*

$\lambda = cT/X$ *Courant number, for ensuring that the discrete scheme works*

$c = \sqrt{T_e/\rho}$ T_e = *string tension*

ρ = *string linear density*

b_1 *air damping coefficient*

b_2 *string internal friction coefficient*

b_H *fluid damping coefficient*

Typical parameter values

	C2	C4	C7	
String				
f_1	52.8221	262.1895	2112.1	Hz
L	1.92	0.62	0.09	m
M_S	35×10^{-3}	3.93×10^{-3}	0.467×10^{-3}	Kg
T_e	750	670	750	N
b_1	0.25	1.1	9.17	s^{-1}
b_2	7.5×10^{-5}	2.7×10^{-4}	2.1×10^{-3}	s
ϵ	7.5×10^{-6}	3.82×10^{-5}	8.67×10^{-4}	
Hammer				
M_H	4.9×10^{-3}	2.97×10^{-3}	2.2×10^{-3}	Kg
p	2.3	2.5	3.0	
b_H	1×10^{-4}	1×10^{-4}	1×10^{-4}	s^{-1}
K	4×10^{8}	4.5×10^{9}	1×10^{12}	
a	0.12	0.12	0.0625	
Boundary				
ζ_l	1×10^{20}	1×10^{20}	1×10^{20}	$\Omega/\mathrm{Kg.m}^{-2}.\mathrm{s}^{-1}$
ζ_b	1000	1000	1000	$\Omega/\mathrm{Kg.m}^{-2}.\mathrm{s}^{-1}$
Sampling				
f_s	4×44.1	4×44.1	4×44.1	kHz
M	521	140	23	

Boundary conditions

- m=1

$$y_m^{n+1} = a_1(y_{m+2}^n - y_m^n + 2y_{m-2}^n) + a_2(y_{m+1}^n + y_{m-1}^n) + a_3 y_m^n + a_4 y_m^{n-1}$$
$$+a_5(y_{m+1}^{n-1} + y_{m-1}^{n-1}) + a_F F_m^n$$

- m=M-1

$$y_m^{n+1} = a_1(2y_{m+1}^n - y_m^n + 2y_{m-2}^n) + a_2(y_{m+1}^n + y_{m-1}^n) + a_3 y_m^n + a_4 y_m^{n-1}$$
$$+a_5(y_{m+1}^{n-1} + y_{m-1}^{n-1}) + a_F F_m^n$$

- m=0

$$y_m^{n+1} = b_{L1} y_m^n + b_{L2} y_{m+1}^n + b_{L3} y_{m+2}^n + b_{L4} y_m^{n-1} + b_{LF} F_m^n$$

- m=M

$$y_m^{n+1} = b_{R1} y_m^n + b_{R2} y_{m-1}^n + b_{R3} y_{m-2}^n + b_{R4} y_m^{n-1} + b_{RF} F_m^n$$

$$b_{R1} = \frac{2 - 2\lambda^2\mu - 2\lambda^2}{1 + b_1 T + \zeta_b \lambda}$$

$$b_{R2} \frac{4\lambda^2\mu + 2\lambda^2}{1 + b_1 T + \zeta_b \lambda}$$

$$b_{R3} = \frac{-2\lambda^2\mu}{1 + b_1 T + \zeta_b \lambda}$$

$$b_{R4} = \frac{-1 - b_1 T + \zeta_b \lambda}{1 + b_1 T + \zeta_b \lambda}$$

$$b_{R5} = \frac{T^2/\rho}{1 + b_1 T + \zeta_b \lambda}$$

for b_{Li} values, just substitute ζ_b with ζ_l

Discrete power law

- g(m, m_0) should be set as a **spatial window** with a **smooth shape** at its boundaries, to avoid for discretize infinite slope boundary problems.
- A possible choice for simulation purposes is a **Hanning window** of (discretized) length 2δ.

$$F_H(n) = K\left|\eta^n - y^n_{m_0}\right|^p \qquad \eta^{n+1} = d_1\eta^n + d_2\eta^{n-1} + d_F F_H(n)$$

$$d_1 = \frac{2}{1 + b_H T/2M_H}$$

$$d_2 \frac{-1 + b_H T/2M_H}{1 + b_H T/2M_H}$$

$$d_F = \frac{-T^2/M_H}{1 + b_H T/2M_H}$$

Initial conditions

- String speed at instant 0: the string is not moving, therefore the speed is zero.

- String displacement at instant 0: the string is in its equilibrium position.

- Hammer speed at instant 0: it is moving, therefore it has a non zero initial value equal to V_{H0}.

- Hammer displacement at instant 0: it is set to be in contact with the string. Its displacement $\eta(0)$ wrt the string equilibrium position is equal to zero.

- The hammer displacement at $n = 1$ depends on the initial velocity V_{H0} and on the temporal resolution between n and $n + 1$ (ΔT), i.e $\eta(1) = V_{H0}\Delta T$. The displacement can be computed, according to the given formula, only from $n = 2$.

- Hammer force at instant 0 is equal to 0, being $\eta(0)$ and $y(m, 0) = 0$.

Bibliography

- ’’Circuit based classical guitar model’’, J. Lee M. French, Applied Acoustics Volume 97, October 2015, Pages 96-103.
- ‘’Numerical simulations of piano strings. I. A physical model for a struck string using finite difference methods’’, A. Chaigne, A. Askenfelt, The Journal of the Acoustical Society of America 95, 1112 (1994).
- ‘’Physical modelling of the piano: An investigation into the effect of string stiffness on the hammer-string interaction’’, C. Saitis, dissertation, Master of Arts in Sonic Arts, Queen’s University Belfast.