



HAL
open science

Nonlinear physical models of vibration and sound synthesis

David Roze, Joël Bensoam

► **To cite this version:**

David Roze, Joël Bensoam. Nonlinear physical models of vibration and sound synthesis. Unfold Mechanics for Sounds and Music, Sep 2014, PARIS, France. pp.1-1. hal-01161074

HAL Id: hal-01161074

<https://hal.science/hal-01161074v1>

Submitted on 8 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Nonlinear physical models of vibration and sound synthesis

David Roze, Joël Bensoam

IRCAM-CNRS UMR 9912 STMS-UPMC, 1 place Igor Stravinsky, F-75004, Paris, France

Abstract—Sound production in musical instruments is the consequence of interactions and wave propagation that include nonlinear phenomena. Simulating these phenomena will enable sound synthesis softwares (such as Modalys, developed at IRCAM) to produce more realistic sounds. In order to do so, nonlinear physical models will be implemented in Modalys software using Green-Volterra kernels.

Green-Volterra kernels are used to simulate space-time nonlinear dynamical problems. This formalism allows to keep the modal approach and simulate nonlinear dynamics until a given order in the Green-Volterra series. This numerical method has been chosen in order to keep "near real-time" computation time. Interactions solving requires to compute the inverse problem, i.e. compute a force using known displacement or velocity. Green-Volterra kernels of a nonlinear string model and interaction definition will be presented with associated numerical results.

I. INTRODUCTION

Modalys is a sound synthesis software based on physical models. The user chooses resonators (string, pipe, plate...) and connect them with interactions (adhere, bow, strike...). The wave propagation in each resonator is computed using linear models, whereas interactions involves nonlinear relations between the force and the displacement or velocity. The aim of this work is to extend Modalys by using nonlinear wave propagation in resonators. A first work has been presented last year in [1] using a string model and the Volterra series. However the use of Volterra series implies a limitation over the spatial repartition of the force which cannot be time-dependent. This is the reason why, the same approach will be used with Green-Volterra series which have been recently defined [2]. A reminder of sound synthesis using linear physical models is made in section II, then a brief definition of the Green-Volterra series is exposed in section III. Finally, section IV describes the computation of interactions between resonators described by their Green-Volterra kernels.

II. LINEAR SOUND SYNTHESIS

A. Green's formalism

If a resonator described by a linear physical model, modal decomposition is one option to solve its dynamics. External forces can be applied using the Green's function. For time invariant problems with $x \in \Omega$ and $t \in \mathbb{R}_+$ displacement is expressed as [3]

$$u(x, t) = \int_{\Omega \times \mathbb{R}_+} g(x; \xi, \tau) f(\xi, t - \tau) d\xi d\tau$$

which gives, in the numerical point of view, discrete instantaneous linear equations

$$u(x_j, t_i) = \tilde{u}(x_j, t_i)_{\rightarrow 0} + \sum_k g(x_j; \xi_k, T) f(\xi_k, t_i) \quad (1)$$

where T is the sampling period and the term $\tilde{u}(x_j, t_i)_{\rightarrow 0}$ determines the state of the system, at time t_i and at point x_j , in the absence of applied forces at the same moment t_i .

If there are m interactions points (with $m \leq N$ the number of points), the wave propagation and interactions are computed by solving the system $\forall j = 1, \dots, m$

$$\begin{cases} u(x_j, t_i) = \tilde{u}(x_j, t_i)_{\rightarrow 0} + \sum_{k=1}^m g(x_j; \xi_k, t_i) f(\xi_k, t_i) \\ f(x_j, t_i) = \mathcal{C}_{(l)}(u(x_j, t_i)), \end{cases}$$

where $\mathcal{C}_{(l)}$ are the given interaction models. If these models are linear, the solution is trivial. If the models $\mathcal{C}_{(l)}$ are nonlinear an iterative Uzawa's algorithm is used.

To compute the needed Green's function for a given linear system the modal formalism is used to provide its decomposition on the modal basis. Interaction will be solved with the modal projections of the Green's function.

B. Structure of simulation

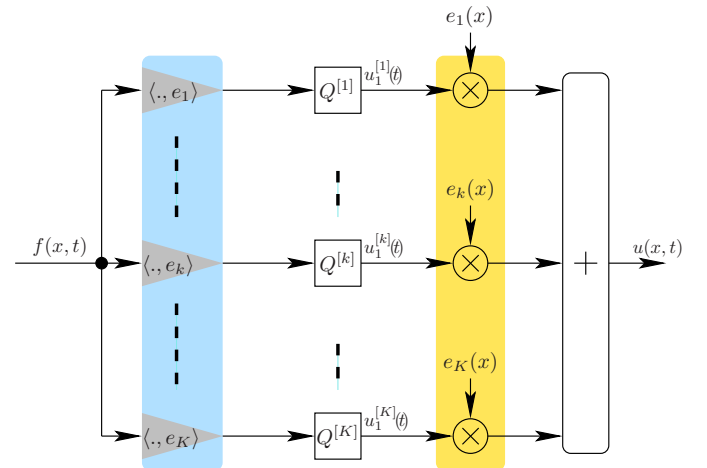


Fig. 1. Linear sound synthesis: the propagation equation of each resonator is projected on its modal basis. Limiting the development in the first predominant modes (in practice, tens or hundreds modes), it allows to obtain a finite dimensional system of recursive filters and provides a numerical representation of the behaviour of a substructure irrespective of its nature: mechanical or acoustic. The set of filters performs the simulation for K modes. Each filter $Q^{[k]}$ computes a modal output $\mathbf{X}_{n+1}^{[k]}$ as a function of $\mathbf{X}_n^{[k]}$ and input $f_{n+1}^{[k]}$.

Given a modal decomposition $u(x, t) = \sum_{k=1}^K u^{[k]}(t)e_k(x)$, a second order boundary value problem, used to describe a wave propagation, can be written as a set¹ of first order differential equations for $k \in \{1, 2, \dots, K\}$

$$\begin{cases} \mathbf{X}^{[k]}(t) = [u^{[k]}(t) \quad \dot{u}^{[k]}(t)]^T \\ \dot{\mathbf{X}}^{[k]}(t) = \mathbf{A}^{[k]}\mathbf{X}^{[k]}(t) + \mathbf{B}f^{[k]}(t) \\ \mathbf{A}^{[k]} = \begin{bmatrix} 0 & 1 \\ -\omega_k^2 & -2c_k \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{cases}$$

where the matrix $\mathbf{A}^{[k]}$ captures the modal data (eigen pulsation ω_k , damping c_k), $f^{[k]}$ are the modal forces. Using the exponential map, a solution can be formulated as

$$\mathbf{X}^{[k]}(t) = \int_0^t e^{\mathbf{A}^{[k]}(t-\tau)} \mathbf{B}f^{[k]}(\tau) d\tau + e^{\mathbf{A}^{[k]}t} \mathbf{X}(0),$$

which gives, after a time discretization: $t_i = iT$ and a zeroth order approximation of the input force, a recursive filter formula

$$\mathbf{X}^{[k]}(t_{i+1}) = e^{\mathbf{A}^{[k]}T} \mathbf{X}^{[k]}(t_i) + \mathbf{B}_0^{[k]} f^{[k]}(t_{i+1}) \quad (2)$$

with $\mathbf{B}_0^{[k]} = -\mathbf{A}^{[k]-1} [\mathbf{B} - e^{\mathbf{A}^{[k]}T} \mathbf{B}]$. A modal reconstruction $\sum_k \mathbf{X}^{[k]} e_k$ leads to a formalism in accordance with Eq. (1). Technically, the computation of the exponential gives rise to a sound synthesis process described in Fig. 1.

C. Interaction

Solving the interaction in one point consists in solving the inverse problem in Eq. (2) after modal reconstruction:

$$\sum_{k=1}^K \mathbf{X}^{[k]}(t_i) e_k(x_j) = \sum_{k=1}^K \left(e^{\mathbf{A}^{[k]}T} \mathbf{X}^{[k]}(t_i) + \mathbf{B}_0^{[k]} f^{[k]}(t_i) \right) e_k(x_j)$$

$$\mathbf{X}(x_j, t_i) = \tilde{\mathbf{X}}(x_j, t_i) + \sum_{k=1}^K \sum_{l=1}^N \mathbf{B}_0^{[k]} f(y_l, t_i) e_k(x_j) e_k(y_l) \Delta y$$

where Δy is the step of space discretization into N points. The term $\sum_{k=1}^K \mathbf{B}_0^{[k]} e_k(x_j) e_k(y_l) dy$ is the definition of the Green's function $g(x_j; y_l, T)$ in the discrete space and time domains.

What would happen if this formalism were extended to the case of nonlinear resonators? The direct problem has been presented recently [2]. The main results will be recalled. Then, on this basis a method to solve interactions will be proposed.

III. GREEN-VOLTERRA SERIES

Nonlinear wave propagation cannot be computed using modal decomposition and Green's operator. In a previous work [1], a study of interaction between nonlinear resonators described by Volterra series has been performed. However, Volterra series cannot be used to solve interaction with a space-varying force distribution, since the Volterra kernels are only time-dependent. In order to compute interactions with unknown force distribution, the kernels need to be a function of space in order to extend the Green's formalism to the nonlinear case. This has been done in [2].

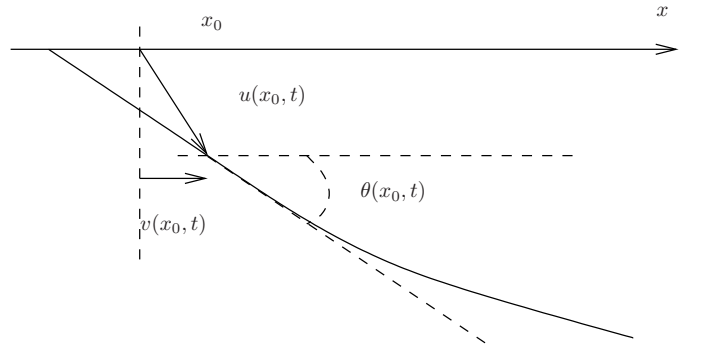


Fig. 2. Transverse and longitudinal displacement u and v and angle θ involve a variation of the string tension which cannot be neglected for large amplitudes.

A. Kirchhoff string model

Equilibrium equation of small element of a string is based on the following relations:

$$\begin{aligned} \frac{\partial}{\partial x} [T(x, t) \sin \theta(x, t)] &= \rho A \frac{\partial^2 u(x, t)}{\partial t^2} \\ \frac{\partial}{\partial x} [T(x, t) \cos \theta(x, t)] &= \rho A \frac{\partial^2 v(x, t)}{\partial t^2} \end{aligned}$$

with ρ the density of the material, A the cross section area, θ the angle between the string axis and the tangent at the considered point (cf. Fig 2), u and v respectively the transverse and longitudinal displacements. In the following v will be neglected.

In the linear string model the tension T is constant and is equal to T_0 the tension of the string at rest. For large amplitude displacements, this assumption does not hold anymore. The tension is therefore defined by

$$\begin{aligned} T(x, t) - T_0 &= EA \left(\sqrt{\left(1 + \frac{\partial v(x, t)}{\partial x}\right)^2 + \left(\frac{\partial u(x, t)}{\partial x}\right)^2} - 1 \right) \\ &\approx EA \left(\sqrt{1 + \left(\frac{\partial u(x, t)}{\partial x}\right)^2} - 1 \right) \end{aligned}$$

In the Kirchhoff string model (see [4], [5]) this expression is averaged over the length, giving the equilibrium equation for $(x, t) \in [0, L] \times \mathbb{R}_+^*$

$$\begin{aligned} \frac{\partial^2 u(x, t)}{\partial t^2} + \delta \frac{\partial u(x, t)}{\partial t} \\ = \left[c^2 + b \int_0^L \left(\frac{\partial u(x, t)}{\partial x}\right)^2 dx \right] \frac{\partial^2 u}{\partial x^2} + f(x, t). \quad (3) \end{aligned}$$

The coefficient δ represents fluid damping. The sound speed is c and $b = \frac{E}{2\rho L}$ is a coefficient of nonlinearity which is a function of the Young's modulus, the density and the length.

¹a modal truncation is performed in order to obtain a finite dimensional set.

B. Definition

A system with input $f(x, t)$ and output $u(x, t)$ is described by a Green-Volterra series of kernels $\{g_n\}_{n \in \mathbb{N}^*}$ if the output is given by

$$u(x, t) = \sum_{n=1}^{+\infty} \int_{(\Omega \times \mathbb{R})^n} g_n(x, t, \xi_{1:n}, \tau_{1:n}) f(\xi_1, \tau_1) \dots f(\xi_n, \tau_n) d\xi_{1:n} d\tau_{1:n} \quad (4)$$

where $\xi_{1:n}$ denotes the vector (ξ_1, \dots, ξ_n) .

C. Green-Volterra kernels

Simulation will not be performed using Eq. (4) since multiconvolution would be too costly in computation time. The expression of the first kernels will be given, then a structure of simulation will be designed using these expressions.

A recurrence relation of the Green-Volterra kernels $\{G_n\}_{n \in \mathbb{N}^*}$ in the Laplace domain can be established for time-invariant system using interconnection laws and cancelling system defined in [2, §3.3 and 3.4]

$$\left[\widehat{s}_{1:n}^2 + \delta \widehat{s}_{1:n} - \partial_x^2 \right] G_n(x; \xi_{1:n}, s_{1:n}) = R_n(x; \xi_{1:n}, s_{1:n})$$

where $\widehat{s}_{1:n}$ is the sum $s_1 + \dots + s_n$ and with

$$\begin{cases} R_1(x; \xi, s) &= \delta_x(\xi) \\ R_n(x; \xi_{1:n}, s_{1:n}) &= b \sum_{\substack{p, q, r \geq 1 \\ p+q+r=n}} \int_{\Omega} \left(\partial_x G_p(x; \xi_{1:p}, s_{1:p}) \right. \\ &\quad \left. \partial_x G_q(x; \xi_{p+1:p+q}, s_{p+1:p+q}) \right) dx \\ &\quad \left. \partial_x^2 G_r(x; \xi_{p+q+1:n}, s_{p+q+1:n}) \right) \end{cases}$$

The first kernel G_1 is the Green's function of the linearized problem defined in the Laplace domain.

These kernels can be decomposed on the orthonormal Hilbert basis $\{e_k\}_{k \in \mathbb{N}^*}$ and it can be shown [2, §4.3] that $\forall n \in \mathbb{N}^*$

$$G_n(x; \xi_{1:n}, s_{1:n}) = \sum_{k=1}^{+\infty} G_n^{[k]}(s_{1:n}) e_{k, \dots, k}^{1+n}(x, \xi_{1:n})$$

with

$$e_{k, \ell_1, \dots, \ell_n}^{1+n}(x, \xi_{1:n}) = [e_k \otimes e_{\ell_1} \otimes \dots \otimes e_{\ell_n}](x, \xi_{1:n}).$$

where \otimes is the tensor product $[e_k \otimes e_l](x, y) = e_k(x)e_l(y)$.

Finally Green-Volterra kernels will be computed by

$$G_n^{[k]}(s_{1:n}) = G^{[k]}(\widehat{s}_{1:n}) R_n^{[k]}(s_{1:n})$$

where $G^{[k]}(s) = (s^2 + \delta s + k^2 \pi^2)^{-1}$ is, for each mode, the transfer function describing the linear part of the model. The source terms are

$$\begin{cases} R_1^{[k]}(s) &= 1 \\ R_n^{[k]}(s_{1:n}) &= \frac{-bk^2 \pi^4}{L^4} \sum_{\substack{p, q, r \geq 1 \\ p+q+r=n}} \sum_{\ell=1}^{+\infty} \ell^2 G_p^{[\ell]}(s_{1:p}) G_q^{[\ell]}(s_{p+1:p+q}) G_r^{[k]}(s_{p+q+1:n}). \end{cases}$$

The first kernels for $n \geq 2$ are

$$G_3^{[k]}(s_{1:3}) = \gamma_k G^{[k]}(\widehat{s}_{1:3}) \sum_{\ell=1}^{+\infty} \ell^2 G^{[\ell]}(s_1) G^{[\ell]}(s_2) G^{[k]}(s_3) \quad (5)$$

$$G_5^{[k]}(s_{1:5}) = \gamma_k G^{[k]}(\widehat{s}_{1:5}) \sum_{\substack{p, q, r \geq 1 \\ p+q+r=5}} \sum_{\ell=1}^{+\infty} \ell^2 G_p^{[\ell]}(s_{1:p}) G_q^{[\ell]}(s_{p+1:p+q}) G_r^{[k]}(s_{p+q+1:5})$$

$$\text{with } \gamma_k = \frac{-bk^2 \pi^4}{L^4}.$$

The first kernels are known, a structure of simulation based on filters will be made. Indeed multiconvolution will be not be considered for sound synthesis with "real-time objectives".

D. Structure of simulation

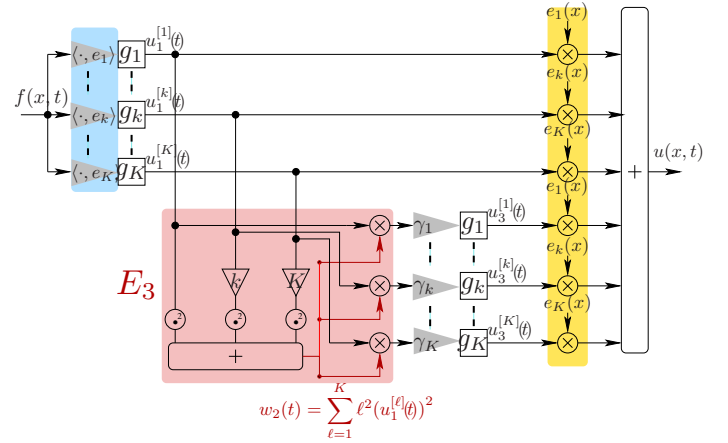


Fig. 3. Structure of simulation of Eq. (3) using Green-Volterra kernels. g_k are identified as the modal projections of the Green's function of the system. $\langle f, g \rangle$ is the scalar product $\int_{\Omega} f(x)g(x)dx$. A first set of filters computes the linear response $u_1^{[k]}$ for each mode. Another set of filters will compute the dynamics of order 3. The input of this set is defined in the block E_3 using Eq. (5).

In [2] and [6] a structures of simulation based on the Volterra or Green-Volterra kernels modal projection have been made. The result for the Kirchhoff string model is presented in Fig. 3 where g_k are the modal projections of the Green's function of the linearized version of equation Eq. (3).

The linear part of the simulation is now performed by

$$\mathbf{X}_1^{[k]}(t_i) = \tilde{\mathbf{X}}_1^{[k]}(t_i) + \mathbf{B}_0^{[k]} f^{[k]}(t_i)$$

$$\text{with } f^{[k]}(t_i) = \langle f, e_k \rangle(t_i) \text{ and } \mathbf{X}_1^{[k]}(t_i) = \begin{bmatrix} u_1^{[k]}(t_i) \\ \dot{u}_1^{[k]}(t_i) \end{bmatrix}.$$

The third order component is computed by

$$\mathbf{X}_3^{[k]}(t_i) = \tilde{\mathbf{X}}_3^{[k]}(t_i) + \mathbf{B}_0^{[k]} f_3^{[k]}(t_i) \quad (6)$$

$$\text{with } f_3^{[k]}(t_i) = \gamma_k u_1^{[k]}(t_i) \sum_{\ell=1}^K \ell^2 (u_1^{[\ell]}(t_i))^2.$$

IV. INTERACTIONS

A. Definition with Green-Volterra series

In order to compute the interaction force, the relation between the force and the displacement/velocity has to be written. Let's begin with the linear part

$$\begin{aligned}\mathbf{X}_1(x, t_i) &= \sum_{k=1}^K \left(\tilde{\mathbf{X}}_1^{[k]}(t_i) + \mathbf{B}_0^{[k]} f^{[k]}(t_i) \right) e_k(x) \\ &= \sum_{k=1}^K \left(\tilde{\mathbf{X}}_1^{[k]}(t_i) + \mathbf{B}_0^{[k]} \int_{\Omega} f(y, t) e_k(y) dy \right) e_k(x) \\ &= \tilde{\mathbf{X}}_1(t_i) + \int_{\Omega} \sum_{k=1}^K \mathbf{B}_0^{[k]} e_k(x) e_k(y) f(y, t) dy \\ &= \tilde{\mathbf{X}}_1(t_i) + \int_{\Omega} \Gamma_{11}(x, y) f(y, t) dy.\end{aligned}$$

Doing the same procedure until order 3 using Eq. (6) leads to

$$\begin{aligned}\mathbf{X}(x, t_i) &= \sum_{k=1}^K \left(\tilde{\mathbf{X}}^{[k]}(t_i) + \mathbf{B}_0^{[k]} (f^{[k]}(t_i) + f_3^{[k]}(t_i)) \right) e_k(x) \\ &= \tilde{\mathbf{X}}_1(t_i) + \tilde{\mathbf{X}}_3(t_i) \\ &+ \iiint_{\Omega^3} \Gamma_3(x, y, z, a) f(y, t_i) f(z, t_i) f(a, t_i) dy dz da \\ &+ \iint_{\Omega^2} \Gamma_2(x, y, z, t_i) f(y, t_i) f(z, t_i) dy dz \\ &+ \int_{\Omega} \Gamma_{11}(x, y) f(y, t_i) dy + \int_{\Omega} \Gamma_{12}(x, y, t_i) f(y, t_i) dy \\ &+ \Gamma_0(x, t_i)\end{aligned}\quad (7)$$

with

$$\begin{aligned}\Gamma_3(x, y, z, a) &= \sum_{k=1}^K \gamma_k \mathbf{B}_0^{[k]} \mathbf{C} \mathbf{B}_0^{[k]} e_k(x) e_k(y) \\ &\quad \sum_{\ell=1}^K \ell^2 (\mathbf{C} \mathbf{B}_0^{[\ell]})^2 e_{\ell}(z) e_{\ell}(a) \\ \Gamma_2(x, y, z, t_i) &= \sum_{k=1}^K \gamma_k \mathbf{B}_0^{[k]} e_k(x) \\ &\quad \left(e_k(z) \sum_{\ell=1}^K 2\ell^2 \tilde{u}_1^{[\ell]}(t_i) \mathbf{C} \mathbf{B}_0^{[\ell]} \mathbf{C} \mathbf{B}_0^{[k]} e_{\ell}(y) \right. \\ &\quad \left. + \sum_{\ell=1}^K \ell^2 (\mathbf{C} \mathbf{B}_0^{[\ell]})^2 \tilde{u}_1^{[k]} e_{\ell}(y) e_{\ell}(z) \right) \\ \Gamma_{12}(x, y, t_i) &= \sum_{k=1}^K \gamma_k \mathbf{B}_0^{[k]} e_k(x) \sum_{\ell=1}^K \ell^2 \tilde{u}_1^{[\ell]}(t_i) \\ &\quad \left(2\mathbf{C} \mathbf{B}_0^{[\ell]} \tilde{u}_1^{[k]}(t_i) e_{\ell}(y) + \tilde{u}_1^{[\ell]}(t_i) \mathbf{C} \mathbf{B}_0^{[k]} e_k(y) \right) \\ \Gamma_0(x, t_i) &= \sum_{k=1}^K \sum_{\ell=1}^K \ell^2 \gamma_k \mathbf{B}_0^{[k]} \tilde{u}_1^{[k]} (\tilde{u}_1^{[\ell]})^2 e_k(x)\end{aligned}$$

It can be seen that finding f consists in solving integral polynomial equations. The discrete version of Eq. (7) leads

to a system of polynomials. The number of variables of these polynomials is related to the number of interaction points.

In this paper, interaction limited to one point will be described in section IV-B and an example of polynomials systems with two point interaction is presented in section IV-C.

B. Example: "Adhere" connection on one point

The result of simulation until nonlinear order N with K modes is given by

$$\mathbf{X}(x_j, t_i) = \sum_{n=1}^N \sum_{k=1}^K \mathbf{X}_n^{[k]}(t_i) e_k(x_j)$$

with

$$\mathbf{X}_n^{[k]}(t_i) = \tilde{\mathbf{X}}_n^{[k]}(t_i) + \mathbf{B}_0^{[k]} f_n^{[k]}(t_i)$$

and can therefore be written as

$$\mathbf{X}(x_j, t_i) = \tilde{\mathbf{X}}(x_j, t_i) + \Pi(f(x_j, t_i)) \quad (8)$$

where Π is a polynomial of order N . Using the discrete version of integration, for $N = 3$, that is

$$\begin{aligned}\Pi(f(x_j, t_i)) &= \Gamma_3(x_j, x_j, x_j, x_j) f^3(x_j, t_i) dx^3 \\ &\quad + \Gamma_2(x_j, x_j, x_j, t_i) f^2(x_j, t_i) dx^2 \\ &\quad + (\Gamma_{12}(x_j, x_j, t_i) + \Gamma_{11}(x_j, x_j)) f(x_j, t_i) dx \\ &\quad + \Gamma_0(x_j, t_i)\end{aligned}$$

Knowing $\tilde{\mathbf{X}}(x_j, t_i)$ and one component of $\mathbf{X}(x_j, t_i)$, solving the interaction consists in solving the polynomial Π at each time step t_i and each interaction point x_j .

An example based on the "Adhere" connection in Modalys can be made with one string. On this string, one point has a zero velocity: this can be seen as one finger pressing the string at this point.

Using the definition of \mathbf{X} in Eq. (8), at the interaction point position x_a , the interaction force f_a is defined by the polynomial

$$0 = \dot{u}(x_a, t_a) + \Pi(f_a). \quad (9)$$

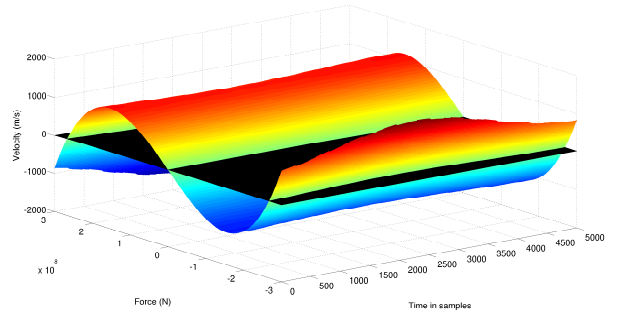


Fig. 4. Interaction law is defined by the black plane: velocity vanishes at the interaction point. The interaction force is obtained by intersection of this law with the dynamics of the nonlinear resonator represented by a polynomial (defined in Eq. (9)) whose order corresponds to the Volterra series truncation order. In case of linear propagation it would be a plane (first order polynomial). With three possible roots, the one with the lowest absolute value is chosen.

The resulting force (cf. Figs. 4 and 5) is then used as an input for the propagation computation of this time step.

Simulation results are presented in section V.

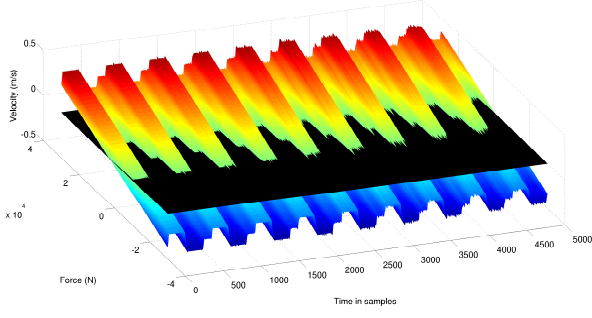


Fig. 5. Zoom of Fig. 4 around the intersection line where the force has the lowest absolute value. It can be seen that the interaction force is alternatively negative and positive.

C. Two points interaction

For interactions involving two points x_a and x_b , Eq. (8) will define \mathbf{X} as a function of interactions forces $f(x_a)$ and $f(x_b)$.

In this case solving the interaction consists in solving

$$\begin{aligned}\dot{u}(x_a, t_i) &= \dot{\hat{u}}(x_a, t_i) + \Pi(x_a, f(x_a, t_i), f(x_b, t_i)) \\ \dot{u}(x_b, t_i) &= \dot{\hat{u}}(x_b, t_i) + \Pi(x_b, f(x_a, t_i), f(x_b, t_i))\end{aligned}$$

with Π defined by

$$\begin{aligned}\Pi(x, f_a, f_b) &= \sum_{k=1}^K e_k(x) \sum_{\ell=1}^K \alpha^{[\ell, k]}(x_a) f_a^3 + 3\beta^{[\ell, k]}(x_a, x_b) f_a^2 f_b \\ &+ 3\beta^{[\ell, k]}(x_b, x_a) f_b^2 f_a + \alpha^{[\ell, k]}(x_b) f_b^3 + \gamma^{[\ell, k]}(x_a) f_a^2 \\ &+ 2\delta^{[\ell, k]}(x_a, x_b) f_a f_b + \gamma^{[\ell, k]}(x_b) f_b^2 + \rho^{[\ell, k]}(x_a) f_a \\ &+ \rho^{[\ell, k]}(x_b) f_b + \Gamma_0(x)\end{aligned}$$

where f_a and f_b are respectively $f(x_a, t_i)$ and $f(x_b, t_i)$.

Polynomial are plotted in Fig. 6. Horizontal axis are interactions forces values in x_a and x_b whereas the vertical axis is the velocity for a given couple of forces. For the case $\dot{u}(x_a, t_i) = \dot{u}(x_b, t_i) = 0$, f_a and f_b are to be found in order to vanish velocity, i.e. solutions will be on the intersection with a black plane $z = 0$ as shown in Fig. 7.

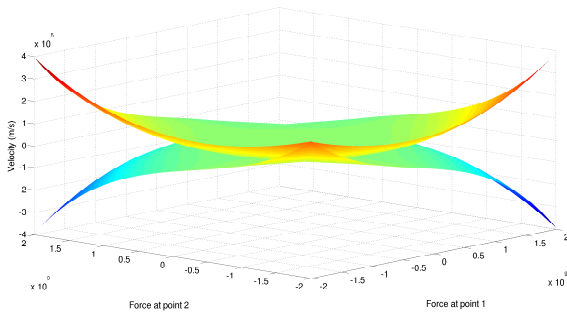


Fig. 6. Polynomials $\dot{\hat{u}}(x_a, t_i) + \Pi(x_a, f(x_a, t_i), f(x_b, t_i))$ and $\dot{\hat{u}}(x_b, t_i) + \Pi(x_b, f(x_a, t_i), f(x_b, t_i))$ at a given time t_i . The value of the two interaction forces is the couple (f_a, f_b) where the point of each surface have the desired velocities $\dot{u}(x_a, t_i)$ and $\dot{u}(x_b, t_i)$.

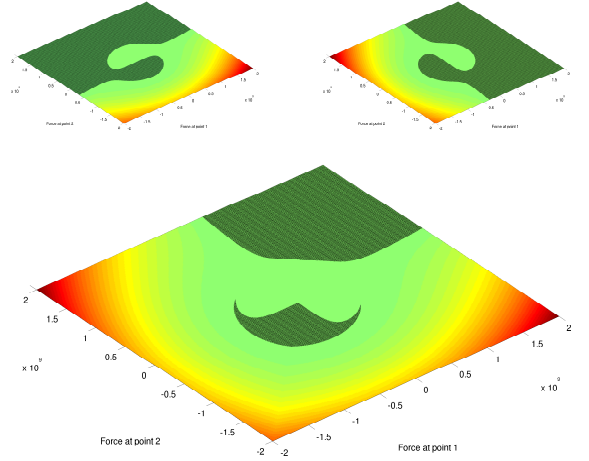


Fig. 7. Upper left: Intersection of the first polynomial with a black plane $z = 0$ defining the interaction (velocity vanishes). The desired forces (f_a, f_b) are on the intersection line between the surface defined by the polynomial and the black plane. Upper right: ibid. with the second polynomial. Bottom: The solutions are the points located on the intersection of the two intersection lines.

For now, the value of the force is determined only with the plots in Fig 7. Then, no simulation has been performed for the two points interaction case.

V. NUMERICAL SIMULATION

A. Constant interaction

A two seconds simulation has been performed for the one point interaction at sample frequency $f_s = 44100\text{Hz}$. The string of length $L = 1.8\text{m}$ is plucked and vibrates freely. At time t_{40000} the interaction is applied at point $x_a = 0.72L$ during 10000 samples and then released. Two simulations have been performed with the same inputs and parameters: one with the linearized version of Eq. (3) i.e. using only the first Green-Volterra kernel G_1 , the other using the two first Green-Volterra kernels G_1 and G_3 since $G_2 = 0$. The spectrograms of the resulting sounds (see Fig. 8) for (a) the linear model and (b) the nonlinear model until order 3 reveal the interaction when frequencies are raised, since the string is shortened. Moreover effects due to the nonlinearity can be seen when vibrations are large enough: on picture (b) energy is transferred to higher modes.

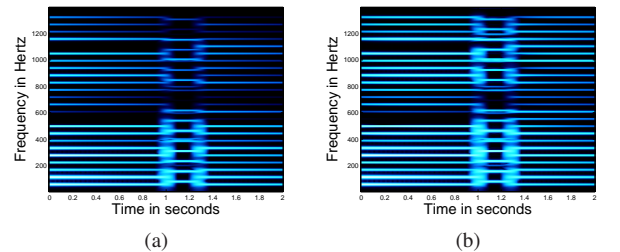


Fig. 8. Spectrograms of the string velocity at point $x_0 = 0.08L$ for (a) the linear model, i.e. using only the first Green-Volterra kernel and (b) using Green-Volterra kernels G_1 and G_3 . At the interaction point $x_a = 0.72L$, the string has a zero velocity during the interaction.

B. Moving interaction

Other simulations have been made with a varying interaction point x_a , shortening the string more and more during the interaction. The resulting frequency variation can be seen in Fig. 9.

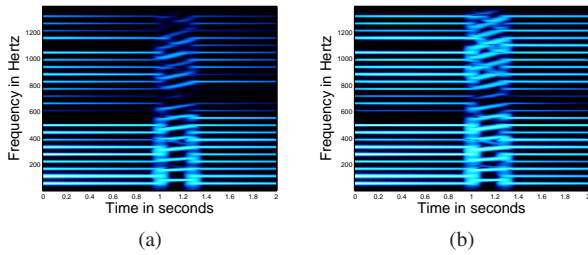


Fig. 9. Spectrograms of the string velocity at point $x_0 = 0.08L$ for (a) the linear model, i.e. using only the first Green-Volterra kernel and (b) using Green-Volterra kernels G_1 and G_3 . During the interaction, the string has a zero velocity at point x_a moving from $0.72L$ to $0.68L$.

This is possible thanks to the use of Green-Volterra series which allows to use a force $f(x, t)$ as an input to the structure of simulation defined in Fig. 3.

C. Convergences issues

However, simulations are not always completed since some parameters may cause divergence in the computations results. This happens when the root with the lowest absolute value is too high, thus giving a interaction force with high value used in the simulation of the nonlinear dynamics. Three parameters have been noticed that can cause these problems:

- the excitation force (and therefore the vibration amplitude) of the string, before the interaction, is too high,
- the duration of the interaction is too long,
- the number of computed modes is too high in comparison with the number of points in the string discretization.

These phenomena will be studied in order to implement interactions with nonlinear physical models, in Modalys software.

VI. CONCLUSION

This work has presented a simple example of sound synthesis with one interaction based on a nonlinear physical model. The simulation performed using Green-Volterra series involves a force $f(x, t)$ located in one point of the resonator.

The use of Green-Volterra series allows to compute nonlinear dynamics at a low cost. The convergence has not been studied and has become important in the case of solving interactions since it can prevent computation to complete. The different cases of divergence need to be delimited in order to implement this method in Modalys software.

However, solving polynomials of order 3 to compute interactions is time consuming. These algorithms will be implemented with a low-level programming language. If computation time is still too high, parallel computing could be a solution to approach “real-time” sound synthesis.

Finally multi-point interaction is a preliminary work that needs to be further investigated.

ACKNOWLEDGEMENT

REFERENCES

- [1] D. Roze and J. Bensoam, “Solving interactions between nonlinear resonators,” in *Sound and Music Computing Conference 2013*, Stockholm, Sweden, July 2013, pp. 576–583.
- [2] D. Roze and T. H elie, “Introducing a Green-Volterra series formalism to solve weakly nonlinear boundary problems: application to the Kirchhoff’s string,” *Journal of Sound and Vibration*, vol. 333, no. 7, pp. 2073–2086, 2014.
- [3] J. Bensoam, “Repr esentation int egrale appliqu ee   la synth ese sonore par mod elisation physique : m ethode des  l ements finis,” Th ese de doctorat, Acad emie de Nantes Universit e du Maine, 2003.
- [4] G. Kirchhoff, *Vorlesungen  ber Mathematische Physik: Mechanik*. Leipzig: Teubner, 1877.
- [5] G. F. Carrier, “On the non-linear vibration problem of the elastic string,” *Quarterly of Applied Mathematics*, vol. 3, pp. 157–165, 1945.
- [6] T. H elie and D. Roze, “Sound synthesis of a nonlinear string using Volterra series,” *Journal of Sound and Vibration*, vol. 314, pp. 275–306, 2008.