



Analog circuits and Port-Hamiltonian realizability issues: a resolution method for simulations via equivalent components

Judy Najnudel, Thomas H elie, David Roze, Henri Boutin, Thierry Maniguet, St ephane Vaiedelich

► To cite this version:

Judy Najnudel, Thomas H elie, David Roze, Henri Boutin, Thierry Maniguet, et al.. Analog circuits and Port-Hamiltonian realizability issues: a resolution method for simulations via equivalent components. 145th AES Convention, Oct 2018, New-York, NY, United States. pp.10053. hal-02297400

HAL Id: hal-02297400

<https://hal.archives-ouvertes.fr/hal-02297400>

Submitted on 27 Sep 2019

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 destin ee au d ep ot et  a la diffusion de documents scientifiques de niveau recherche, publi es ou non,  emanant des  tablissements d'enseignement et de recherche franais ou  trangers, des laboratoires publics ou priv es.

Analog circuits and Port-Hamiltonian realizability issues: a resolution method for simulations via equivalent components.

Judy Najnudel², Thomas Hélie², Henri Boutin², David Roze²,
Thierry Maniguet¹, and Stéphane Vaiedelich¹

¹CNRS-Musée de la Musique
²STMS-S3AM

Abstract

In order to simulate the Ondes Martenot, a classic electronic musical instrument, we aim to model its circuit using Port-Hamiltonian Systems (PHS). PHS have proven to be a powerful formalism to provide models of analog electronic circuits for audio applications, as they guarantee the stability of simulations, even in the case of non-linear systems. However, some systems cannot be converted directly into PHS because their architecture cause what are called realizability conflicts. The Ondes Martenot circuit is one of those systems. In this paper, a method is introduced to resolve such conflicts automatically: problematic components are replaced by equivalent components without altering the overall structure nor the content of the modeled physical system.

1 Introduction

The Ondes Martenot, invented by Maurice Martenot in 1928, are one of the first electronic musical instruments [1]. Though the instrument was widely popular during the last century and still has an important repertoire and community, its production stopped in 1988. Due to the obsolescence of its components, an accurate simulation would be highly relevant for musicians, composers and musicologists. Port-Hamiltonians systems (PHS) have proven to be a powerful approach to simulate analog electronic circuits for audio applications [2], as they guarantee passivity of systems and the stability of simulations. Introduced in the 1990's [3], PHS describe open dynamical systems and their inherent interactions as a network of energy storage components, dissipative components, and

connection ports. It is multi-physical (a system can be electrical, mechanical, thermal or a mix as well), modular (a system made of several PHS is still a PHS) and passive (no energy is spontaneously created). However, this formalism is sensitive to realizability conflicts and thus not every circuit can be directly modeled via this approach. Those conflicts appear for instance (but not only) in presence of parallel capacitors; the Ondes Martenot circuit happens to be subject to such conflicts as it contains several parallel capacitors. As the laws of those capacitors are not well known yet, we choose not to impose linear behaviors and as a consequence, a common impedance analysis approach cannot apply. We thus present a method resolving realizability issues by automatically replacing components causing conflicts with equivalent components, provided that the laws of the replaced components can be represented by piecewise linear scalar functions.

This paper is structured as follows: in section 2 we briefly present the PHS formalism and a corresponding numerical scheme for simulations. In section 3, we give the description of a realizability conflict, and introduce a method to resolve it without losing information about the physical behavior of the system for mono-variate components. We then suggest a way of implementing this method within the open-source library PyPHS. Finally, after introducing the Ondes Martenot circuit in section 4, we present the results of our method implementation in section 5.

2 Port-Hamiltonian Systems: formalism and example

2.1 Formalism

Consider a physical system as a network of storage components with their state variable \mathbf{x} and their stored energy given by a constitutive law $H(\mathbf{x})$, dissipative components with their variable \mathbf{w} and dissipative law $z(\mathbf{w})$, control inputs \mathbf{u} and their associated outputs \mathbf{y} . The variables are generally time-dependent and can be multivariate. If such a system is realizable, the energy transfers i.e. flow and effort exchanges between the system components can be expressed into an algebro-differential form using a skew-symmetric matrix \mathbf{S} :

$$\underbrace{\begin{pmatrix} \frac{dx}{dt} \\ w \\ -y \end{pmatrix}}_{\mathcal{F}(\text{flows})} = \mathbf{S} \times \underbrace{\begin{pmatrix} \nabla H(x) \\ z(w) \\ u \end{pmatrix}}_{\mathcal{E}(\text{efforts})} \quad (1)$$

The skew-symmetry of \mathbf{S} guarantees that the system remains passive. Indeed, the power balance requires that

$$\underbrace{\frac{dE}{dt}}_{\nabla H(x)^\top \frac{dx}{dt}} = \underbrace{P_{ext}}_{u^\top y} - \underbrace{P_{diss}}_{z(w)^\top w \geq 0} \quad (2)$$

where $\frac{dE}{dt}$ is the total energy variation of the system, P_{ext} the incoming power from external sources and P_{diss} the total dissipated power which must be positive. Indeed,

$$\begin{aligned} \frac{dE}{dt} + P_{diss} - P_{ext} &= \mathcal{E}^\top \mathcal{F} \\ &= \mathcal{E}^\top S \mathcal{E} \\ &= (\mathcal{E}^\top S \mathcal{E})^\top \\ &= \mathcal{E}^\top S^\top \mathcal{E} \end{aligned} \quad (3)$$

Thus if and only if S is skew-symmetric,

$$\mathcal{E}^\top S \mathcal{E} = -\mathcal{E}^\top S^\top \mathcal{E} = -\mathcal{E}^\top S \mathcal{E} = 0 \quad (4)$$

and the power balance is preserved.

2.2 Example

Consider a parallel RLC circuit (fig.1). The capacitor C and the inductor L

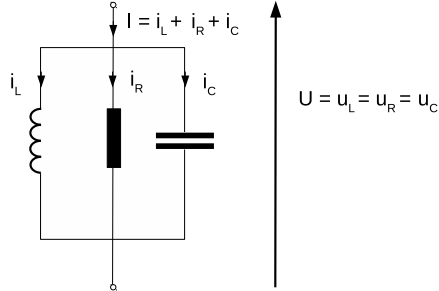


Figure 1: Parallel RLC

are storage components whose states are given by the variables q (charge) and ϕ (magnetic flow) respectively ; the resistor R is a dissipative component described by Ohm's law. The system is current (flow) controlled, its associated output is a voltage (effort). Table 1 recaps the variables and the associated laws of the three components. Using the PHS formalism, the energy transfers (Kirchoff's law) in a parallel RLC circuit take a very simple form involving a skew-symmetric matrix S :

$$\begin{pmatrix} i_C \\ u_L \\ u_R \\ -U \end{pmatrix} = \begin{pmatrix} 0 & -1 & -1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} u_C \\ i_L \\ i_R \\ I \end{pmatrix}$$

	x	$\frac{dx}{dt}$	$H(x)$	$\nabla H(x)$
C	q	$\dot{q} = i_C$	$\frac{q^2}{2C}$	$\frac{q}{C} = u_C$
L	ϕ	$\dot{\phi} = u_L$	$\frac{\phi^2}{2L}$	$\frac{\phi}{L} = i_L$
		w		$z(w)$
R		u_R		$\frac{u_R}{R} = i_R$

Table 1: State variables and constitutive laws for a parallel RLC circuit

Currents are flows, voltages are efforts and their products therefore are powers, thus equation (2) is naturally retrieved. Note that in this simple example, the matrix S is sparse with constant coefficients, but the properties of the PHS formalism hold for non-linear, coupled or multivariate systems which yield more complex matrices.

2.3 Numerical scheme

The PHS formalism guarantees the passivity of the system in continuous time. Introducing discrete gradient [4] in a well chosen numerical scheme allows to preserve this passivity property in discrete time, therefore granting the stability of the simulation as well. A numerical scheme computes the current sample x_{n+1} from the previous samples $x_{n-i}, i \in [0, n]$ and the given parameters of the problem. Here we use a one-step numerical scheme ($i = 0$) which yields

$$x_{n+1} = x_n + \delta x_n(t_s) \quad (5)$$

where t_s is the sampling period. The discrete gradient $\bar{\nabla}H(x_n, \delta x_n)$ is defined by

$$\begin{aligned} \bar{\nabla}H(x_n, \delta x_n) &= \frac{H(x_{n+1}) - H(x_n)}{x_{n+1} - x_n} \\ &= \frac{H(x_n + \delta x_n) - H(x_n)}{\delta x_n} \end{aligned} \quad (6)$$

so that the discrete variation of energy is retrieved by chain derivation:

$$\sigma_n(E_n, t_s) = \bar{\nabla}H(x_n, \delta x_n)^\top \cdot \frac{\delta x_n(t_s)}{t_s} \quad (7)$$

where $\sigma_n(E_n, t_s) = \frac{E_{n+1} - E_n}{t_s}$. Therefore the numerical scheme composed of (5) and (6) guarantees the power balance in discrete time. Here the given data are the matrix S , the laws H and z and the sequence of discrete inputs u_n . The simulation is computed by replacing $\frac{dx}{dt}$ by $\frac{\delta x_n(t_s)}{t_s}$ and $\nabla H(x)$ by $\bar{\nabla}H(x_n, \delta x_n)$ in equation (1) which yields a dynamic equation of the form

$$\delta x_n = t_s f_n(x_n + \alpha \delta x_n), \alpha \in [0, 1] \quad (8)$$

where f_n is a function derived from the matrix S coefficients, $\bar{\nabla}H$, $z(w)$ and u_n . This is an implicit Euler scheme which can be solved using a Newton-Raphson algorithm [5]. Once δx_n is computed, x_{n+1} and y_n can be computed as well.

3 Realizability issues in circuit modeling and resolution

3.1 Problem description

Consider a system composed of at least two connected components, each subject to the same flow if connected in series, or the same effort if connected in parallel. If both components are effort-controlled and connected in series, or flow-controlled and connected in parallel, then the system cannot be described using the PHS formalism. This is the case of two capacitors connected in parallel, a configuration frequently found in analog circuits to refine the value of the equivalent capacity (fig.2). Both capacitors are flow-controlled ($\frac{dx}{dt} = \dot{q} = i$).

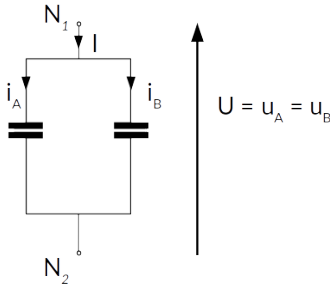


Figure 2: Capacitors connected in parallel

When connected in parallel, their corresponding efforts are equal and no matrix can represent the energy transfers in the PHS formalism :

$$\begin{pmatrix} i_A \\ i_B \\ -U \end{pmatrix} = ? \times \begin{pmatrix} u_A \\ u_B \\ I \end{pmatrix}$$

Such a conflict would appear with two inductors connected in series as well. A criterion for realizability analysis is provided by Falaize [6], involving the graph representation of the circuit. In such a representation, a node is a connection between two components and an edge is the component itself. As a convention, the orientation of an edge is given by the current going through it. The criterion derives from the incidence matrix Γ defined as follows:

$$\Gamma_{n,e} = \begin{cases} -1 & \text{if edge } e \text{ is outgoing node } n \\ 1 & \text{if edge } e \text{ is ingoing node } n \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

The edges are supposed to be sorted between flow-controlled components and effort-controlled components. Then from this incidence matrix Γ , we can extract the matrix γ of the flow-controlled components only. The criterion establishes that if γ is invertible, the system is realizable as a PHS structure. Indeed, for the parallel capacitors example, the γ matrix would yield

$$\gamma = \begin{pmatrix} C_A & C_B \\ -1 & -1 \\ 1 & 1 \end{pmatrix} \begin{matrix} N_1 \\ N_2 \end{matrix} \quad (10)$$

As its determinant equals 0, γ is not invertible and the realizability issue is confirmed.

3.2 Formal resolution for mono-variate components

Such a criterion is convenient to predict if a given system will be realizable or not. But it does not provide a solution in the event of a realizability issue. The idea we propose is to replace non-realizable subsystems with equivalent components without altering the power balance of the system. In the case of linear components, such conflicts pose no difficulty: the association of the two components A and B can be replaced by an equivalent component C whose value is the sum of the previous two, either manually or automatically by graph analysis. Indeed for the parallel capacitors example, if we denote Z each component's impedance, $Z_C = \frac{Z_A Z_B}{Z_A + Z_B}$ leads to $C_C = C_A + C_B$. This is also immediate for two inductors in series with $Z_C = Z_A + Z_B$. The method we present here compute equivalent components for non-linear components as well, provided that their constitutive laws satisfy the right hypotheses. We denote C the equivalent component, q_C its state variable and H_C its constitutive law. H_A and H_B are the constitutive laws of components A and B and we denote their gradients H'_A and H'_B . Our hypotheses are:

(i) H_A et H_B are positive definite and C^1

(ii) H'_A and H'_B are increasing

(iii) $H'_A(0) = H'_B(0) = 0$

H'_A and H'_B are thus bijective which grants H'^{-1}_A and H'^{-1}_B existence.

By definition,

$$u_A = H'_A(q_A) \quad (11)$$

and

$$u_B = H'_B(q_B) \quad (12)$$

which leads to

$$q_A = H'^{-1}_A(u_A) \quad (13)$$

and

$$q_B = H'^{-1}_B(u_B) \quad (14)$$

The components A and B are extensive thus

$$q_C = q_A + q_B \quad (15)$$

and the configuration of the circuit gives

$$u_C = u_A = u_B \quad (16)$$

(or $i_C = i_A = i_B$ for components in series).

We denote

$$\tilde{H}_A(u_A) = (H_A \circ H_A'^{-1})(u_A) = (H_A \circ H_A'^{-1})(u_C) \quad (17)$$

and

$$\tilde{H}_B(u_B) = (H_B \circ H_B'^{-1})(u_B) = (H_B \circ H_B'^{-1})(u_C) \quad (18)$$

The energy of the equivalent component is the sum of the energies of the components A and B and we define $\tilde{H}_C(u_C)$ as:

$$\begin{aligned} \tilde{H}_C(u_C) &= \tilde{H}_A(u_C) + \tilde{H}_B(u_C) \\ &= (\tilde{H}_A + \tilde{H}_B)(u_C) \end{aligned} \quad (19)$$

Equations (15) and (16) yield

$$\begin{aligned} q_C &= H_A'^{-1}(u_A) + H_B'^{-1}(u_B) \\ &= H_A'^{-1}(u_C) + H_B'^{-1}(u_C) \\ &= (H_A'^{-1} + H_B'^{-1})(u_C) \end{aligned} \quad (20)$$

$H_A'^{-1}$ and $H_B'^{-1}$ are bijective so their sum is also bijective. Equations (20) and (19) thus yield

$$\begin{aligned} \tilde{H}_C(u_C) &= (H_A \circ H_A'^{-1} + H_B \circ H_B'^{-1})(u_C) \\ &= (H_A \circ H_A'^{-1} + H_B \circ H_B'^{-1}) \circ (H_A'^{-1} + H_B'^{-1})^{-1}(q_C) \end{aligned} \quad (21)$$

We thus define $H_C(q_C) = (H_A \circ H_A'^{-1} + H_B \circ H_B'^{-1}) \circ (H_A'^{-1} + H_B'^{-1})^{-1}(q_C)$ and this can be generalized for N connected components:

$$H_C(q_C) = \left[\sum_{i=1}^N H_i \circ H_i'^{-1} \right] \circ \left[\sum_{i=1}^N H_i'^{-1} \right]^{-1}(q_C) \quad (22)$$

Additionally, we are able to compute the state variable of the original components at any time:

$$\begin{aligned} q_i &= H_i'^{-1}(u_C) \\ &= (H_i'^{-1} \circ H_C'^{-1})(q_C) \end{aligned} \quad (23)$$

3.3 Implementation in the PyPHS library for mono-variate piecewise linear components

3.3.1 PyPHS library

PyPHS is a Python open-source library dedicated to PHS modeling and simulation [7]. The user defines a system either manually or with a netlist of components

belonging to a dictionary. Each dictionary lists usual components of a physical class (electrical, mechanical, thermal ...) and encodes their constitutive laws. PyPHS then creates an associated graph object and performs a realizability analysis. If the system is realizable, a PHS structure and corresponding equations are derived from the graph and one can compute simulations using the numerical scheme introduced in section 2.3. The complete library and documentation are available on <https://github.com/pyphs/pyphs>.

3.3.2 Construction of components with piecewise linear constitutive laws

In order to solve realizability conflicts using the method presented in section 3.2, we need a class of constitutive laws for which sum, inversion and composition exist and are computable. Piecewise linear functions are thus considered, and we define the H'_i of the form $f_{X,Y} \in C^0(\mathbb{R}, \mathbb{R})$ with $(X, Y) \in \mathbb{I}_N^2$, $N > 0$ where $\mathbb{I}_N = \{(X_0, \dots, X_N) \in \mathbb{R}^{N+1} \text{ s.t. } X_0 < X_1 < \dots < X_N\}$:

(i) if $X_n \leq x < X_{n+1}$ with $n \geq 0$ and $n < N$, then $f_{X,Y}(x) = Y_n + \frac{Y_{n+1}-Y_n}{X_{n+1}-X_n}(x - X_n)$

(ii) if $x < X_0$, then $f_{X,Y}(x) = Y_0 + \frac{Y_1-Y_0}{X_1-X_0}(x - X_0)$

(iii) if $x > X_N$, then $f_{X,Y}(x) = Y_{N-1} + \frac{Y_N-Y_{N-1}}{X_N-X_{N-1}}(x - X_N)$ (iv) it exists n s.t. $0 \leq n < N$ and $X_n = Y_n = 0$. For this class of functions, we define inverse, sum and composition laws as internal laws:

$$f_{X,Y}^{-1} = f_{Y,X}$$

$$f_{A,B} + f_{C,D} = f_{E,F} \text{ where } E = \text{sort}(A, C) \text{ and } F = f_{A,B}(E) + f_{C,D}(E)$$

$$f_{C,D} \circ f_{A,B} = f_{E,F} \text{ where } E = \text{sort}(A, f_{A,B}^{-1}(C)) \text{ and } F = [f_{C,D} \circ f_{A,B}](E)$$

We define $\text{sort}(A, B)$ as the set of all the elements of A and B, sorted. The H'_i are computed from the H_i by integration with trapezoidal rule and laws for equivalent components are given by equation (22). The components replacement algorithm is presented in alg. 1.

4 The Ondes Martenot circuit

The Ondes Martenot, invented by Maurice Martenot in 1928, are one of the first electronic musical instrument based on heterodyne processing: two oscillators generate a high frequency sinusoidal voltage (around 80 kHz); one is fixed, and the player controls the frequency of the second one using a ribbon, whose position controls a capacitor made of several capacitors connected in parallel. The sum of these two voltages is an amplitude-modulated signal whose envelope is detected using a triode vacuum tube, producing an audible sound. The triode vacuum tube is a non-linear component which produces harmonics. This enriched signal is then routed towards special kinds of loudspeakers selected by the musician. Figure 3 shows the schematic of the controllable oscillator of Onde 169. This part of the circuit is a good candidate for our method as it contains several parallel capacitors.

Divide system graph between parallel and series subgraphs;

```

for each subgraph do
  if subgraph is parallel then
    for each storage component do
      if component is flow-controlled then
        | Store component dataset;
      end
    end
    if there are at least two stored components then
      | Compute equivalent  $H$ ;
      | Remove stored components from subgraph;
      | Replace with new component dataset;
    end
  end
  if subgraph is series then
    for each storage component do
      if component is effort-controlled then
        | Store component dataset;
      end
    end
    if there are at least two adjacent stored components then
      | Compute equivalent  $H$ ;
      | Remove stored components from subgraph;
      | Replace with new component dataset;
    end
  end
end

```

Algorithm 1: Component replacement algorithm based on a dataset $(X, Y) \in \mathbb{I}_N^2$ and an interpolating function $f_{X,Y} \in \mathcal{C}^0(\mathbb{R}, \mathbb{R})$

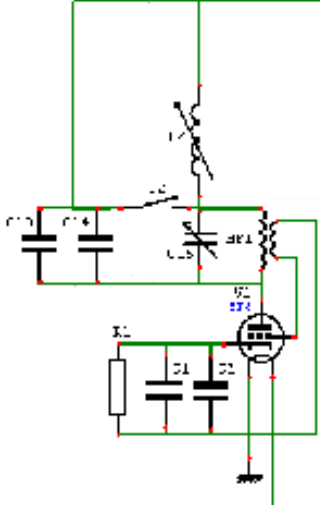


Figure 3: Schematic of the controllable oscillator of Ondes 169 (source: Musée de la Musique, Paris)

5 Results

The algorithm is performed on a system made of three non-linear capacitors in parallel. As the real laws of the circuit capacitors are not known yet since the components could not be measured, we generate data for three piecewise linear functions representing H'_1 , H'_2 and H'_3 with the constraint that $H'_1(X_1) = H'_2(X_2) = H'_3(X_3)$ and take a set X_1 of sorted q values evenly spaced by $0.5 \cdot 10^{-5}$ in $[0, 10^{-4}]$ (see table 2). The manually computed law for such a system is $H(x) = \frac{x^4}{4C^3}$ where $C = \sqrt[3]{C_1} + \sqrt[3]{C_2} + \sqrt[3]{C_3}$. Figure 4 shows a comparison with this law and the one automatically computed using algorithm 1 and equation (22).

	C_1	C_2	C_3
value	440pF	47pF	27pF
X = {q}	X_1	$X_2 = X_1 \sqrt[3]{\frac{C_2}{C_1}}$	$X_3 = X_1 \sqrt[3]{\frac{C_3}{C_1}}$
Y = H'(X)	$\frac{X_1^3}{C_1}$	$\frac{X_2^3}{C_2}$	$\frac{X_3^3}{C_3}$

Table 2: Data generated for three non-linear capacitors in parallel

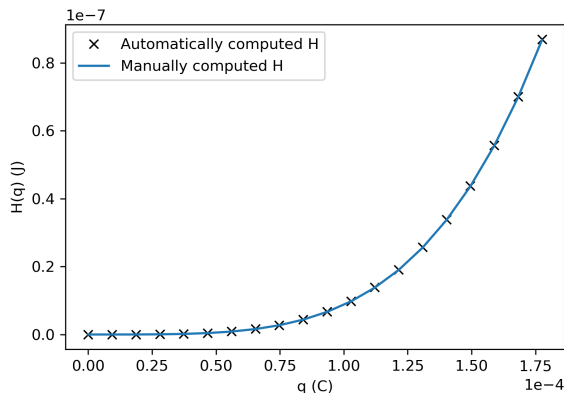


Figure 4: Comparison between manually computed law and automatically computed law for three non-linear capacitors connected in parallel

6 Discussion

The results of the simulation are consistent with the theory. The replacement method allow to compute equivalent laws for non linear mono-variate components connected in a non-realizable way. A modeling based on the PHS formalism can then (1) simulate a larger class of circuits without preprocessing or reorganizing from the user's part and (2) include more complex models as constitutive laws. A current work tackles the extension of this method to multivariate laws, as some components depend from several variables, like the ribbon-controlled capacitor of the Ondes Martenot whose capacity varies in time.

7 Summary

As a summary, this paper shows that some components connections in circuits are not readily adapted to the formalism of Port-Hamiltonian Systems. A simple method to resolve such conflicts of realizability is introduced. This work forms part of larger project, aiming to simulate the whole electronics of Ondes Martenot, which is an essential step to provide reliable sound synthesis.

8 Acknowledgments

This work was funded by the Collegium Musicae and is supported by the project ANR-16-CE92-0028, entitled Interconnected Infinite-Dimensional systems for Heterogeneous Media, INFIDHEM, financed by the French National Research Agency (ANR). Further information is available at <https://websites.isae-supaero.fr/infidhem/the-project>. The authors would also like to par-

ticularly thank Antoine Falaize, post-doctoral fellow, and Tristan Lebrun, PhD candidate.

References

- [1] Leipp, E., “Les Ondes Martenot,” *bulletin du GAM no60*, 1972.
- [2] Falaize, A. and Hélie, T., “Simulation of an analog circuit of a wah pedal: a port-Hamiltonian approach,” in *135th convention of the Audio Engineering Society*, pp. –, New-York, United States, 2013, cote interne IRCAM: Falaize13a.
- [3] Maschke, B. M., Van der Schaft, A. J., and Breedveld, P., “An intrinsic Hamiltonian formulation of network dynamics: Non-standard Poisson structures and gyrators,” *Journal of the Franklin institute*, pp. 923–966, 1992.
- [4] Itoh, T. and Abe, K., “Hamiltonian-conserving discrete canonical equations based on variational difference quotients,” *Journal of Computational Physics*, 76(1), pp. 85–102, 1988.
- [5] Kelley, C., “Iterative methods for linear and nonlinear equations,” *Frontiers in applied mathematics*, 16, pp. 575–601, 1995.
- [6] Falaize, A., *Modélisation, simulation, génération de code et correction de systèmes multi-physiques audios: Approche par réseau de composants et formulation Hamiltonienne à Ports*, Ph.D. thesis, Université Pierre & Marie Curie-Paris 6, 2016.
- [7] Falaize, A. and Hélie, T., “PyPHS: Passive modeling and simulation in python,” 2016, python package, Web page.