Elementary tools on Port-Hamiltonian Systems with applications to audio/acoustics

Thomas Hélie

Lecture notes for the 2nd Spring School on Theory and Applications of

Port-Hamiltonian Systems, Fauenchiemsee, 20-25 March 2022

Elementary tools on Port-Hamiltonian Systems with applications to audio/acoustics

Thomas HÉLIE

S3AM team, STMS laboratory (IRCAM-CNRS-SU), Paris, France

March 2022

This document presents elementary tools on Port-Hamiltonian Systems (PHS) dedicated to the energy-consistent modelling and simulation of linear or nonlinear open physical systems. It focuses on the case of finite-dimensional input-state-output representations of systems constructed as an assembly of multi-physical lumped-components, which limits the prerequisites to differential calculus and matrix algebra. The presentation and exercises are chosen with the aim of providing a toolbox to address audio-acoustic problems with real time application perspectives.

1 Introduction

Port-Hamiltonian Sytems (PHS) introduced by A. van der Schaft and B. Maschke in [1] (see also [2, 3, 4]) describe open energetically-balanced physical systems in a structured form, which naturally encodes the power balance into conservative, dissipative and external parts. This form guarantees passivity and provides a modular composable framework, passivity being inherited through composition of PHS. This document presents elementary tools on linear and nonlinear Port-Hamiltonian Systems (PHS) for the modelling and simulation, that can be used in some audio/acoustic applications.

Why PHS for musical audio/acoustic applications ?

Musical instruments and audio systems are multiphysical systems, which naturally fulfill a power balance. Most of them involve nonlinearities responsible for the timbre evolution versus the excitation amplitude and for self-oscillations (e.g. bowed instruments, wind instruments, voice). Power-balanced numerical methods naturally preserve passivity, avoiding numerical instability artifacts that are often a stumbling block in nonlinear systems. Port-Hamiltonian systems provide a power-balanced and component-based approach, with a modular framework similar to that experienced by

instrument makers or electronics engineers in their workshop: choose, build, refine your components and assemble them. Another important issue is control (mentioned but not addressed in this document), in order to design correctors for audio-transducers, and augmented or hybrid musical instruments (with power-balanced reprogrammed physics).

This document focuses on the modelling (linear and nonlinear) and simulation, introducing tools based on differential calculus and matrix algebra.

Some of these audio/acoustic applications concern (non-exhaustive list):

- **Nonlinear analog audio circuits:** wah pedal [41], ondes Martenot [62, 36, 37], specific components such as operational amplifiers [67] and nonlinear coils [72, 38, 75], circuit graph generation tools for explicit [33] or implicit [71] representations, and identification of nonlinear components in circuit [76],
  (see also the PhDs [27, 30])

- **Electro-acoustics:** loudspeaker [42, 44, 39], electro-mechanic piano [46, 35],
  (see also the PhDs [27, 28])

- **Vibration mechanics:** Duffing oscillator [54], nonlinear strings and membranes [52, 58] with, in the case of the violin, friction modelling [74], tom-drum [68, 70], and generalised (linear and nonlinear) damping models [40, 32, 48, 51],
  (see also the PhD [29])

- **Fluid-Structure interaction and acoustics:** Brass instruments [43, 53, 34] and vocal apparatus (larynx and vocal tract) [56, 63, 59, 66, 65, 69, 73],
  (see also the PhDs [26, 31])

- **Control applications:** loudspeaker correction with differential-flatness derived from PHS [50], passive finite-time controlled systems for a mechanical oscillator [60], for acoustic absorbers [64] and for hybrid tom-drums [68, 70], real-time passive digital controller robust to computational latency [78],
  (see also the PhDs [27, 28, 29]),

- **Numerical methods and automatic code generation:** power-balanced numerical scheme with non-iterative solver based on quadratisation method [49], passive integration scheme of skew-gradient systems [61], automatic code generation with PyPHS library [77] (see also [57, 5]), anti-aliasing for discrete-gradient method [55] and regular power-balanced method [30, chap. 5],
  (see also the PhDs [26, 27, 30]).

It is worth mentioning that PHS and quadratisation methods have recently started to be used for musical applications in other laboratories (see [6, 7]). See also [8] for results on fluid-structure interactions and voice.

Throughout this document, suggestions for exercises are indicated by the symbol $\triangleright$.

Contents

Note: some elements of section 2 are based on an article in preparation.

2 Input-state-output representations of port-Hamiltonian systems

This section presents standard structured state-space representations of PHS in the finite-dimensional case, according to several formulations: (§ 2.1) a *component-based semi-explicit* differential-algebraic formulation, (§ 2.2) a *commonly used* differential formulation. Then, relations between them (§ 2.3-§ 2.4), *interconnections* of PHS (§ 2.5-2.6) and PHS shifting (§ 2.7).

2.1 Differential-Algebraic formulation (semi-explicit DAF)

In this formulation, an open multi-physical system is represented by a network of

(i) *storage components* of state $\mathbf{x}$ and energy $E := H(\mathbf{x}) \geq 0$,

(ii) *memoryless passive components* described by an effort law $\mathbf{z}(\mathbf{w})$ for flow $\mathbf{w}$, such that the *dissipated power* is $P_{\mathbf{diss}} := \mathbf{z}(\mathbf{w})^\intercal \mathbf{w} \geq 0$ ($P_{\mathbf{diss}} = 0$ for conservative ones);

(iii) *external components* receiving power $P_{\mathbf{ext}} := \mathbf{u}^\intercal \mathbf{y}$ through ports represented by[1] system inputs $\mathbf{u}$ and outputs $\mathbf{y}$,

the *internal flows* $\mathbf{f}(t)$ and *efforts* $\mathbf{e}(t)$ of which are coupled according to the following differential-algebraic equation

$$\underbrace{\begin{bmatrix}\dot{\mathbf{x}}\\ \mathbf{w}\\ \mathbf{y}\end{bmatrix}}_{=:\,\mathbf{f}} = \underbrace{\begin{bmatrix}\boldsymbol{S}_{\mathbf{xx}} & \boldsymbol{S}_{\mathbf{xw}} & \boldsymbol{S}_{\mathbf{xu}}\\ * & \boldsymbol{S}_{\mathbf{ww}} & \boldsymbol{S}_{\mathbf{wu}}\\ * & * & \boldsymbol{S}_{\mathbf{yu}}\end{bmatrix}}_{=:\,\boldsymbol{S}=-\boldsymbol{S}^\intercal} \underbrace{\begin{bmatrix}\nabla H(\mathbf{x})\\ \mathbf{z}(\mathbf{w})\\ \mathbf{u}\end{bmatrix}}_{=:\,\mathbf{e}} \quad \begin{array}{ll}(i) & \text{storage} \to \text{differential eq.}\\ (ii) & \text{memoryless} \to \text{algebraic eq.}\\ (iii) & \text{ports} \to \text{physical signals}\end{array} \tag{1}$$

through a skew-symmetric interconnection matrix $\boldsymbol{S}$ possibly depending on $(\mathbf{x}, \mathbf{w}, \mathbf{u})$. Then, the powers received by (i-iii) are balanced:

$$P_{\mathbf{stored}} + P_{\mathbf{diss}} + P_{\mathbf{ext}} = 0, \text{ where } P_{\mathbf{stored}} := \dot{E} \ \ (= \nabla H(\mathbf{x})^\intercal \dot{\mathbf{x}}) \tag{2}$$

denotes the power received by energy-storing components. Indeed, $P_{\mathbf{stored}} + P_{\mathbf{diss}} + P_{\mathbf{ext}} = \mathbf{e}^\intercal \mathbf{f} \overset{(1)}{=} \mathbf{e}^\intercal \boldsymbol{S} \mathbf{e}$ is zero since $\mathbf{e}^\intercal \boldsymbol{S} \mathbf{e} = (\mathbf{e}^\intercal \boldsymbol{S} \mathbf{e})^\intercal = -(\mathbf{e}^\intercal \boldsymbol{S} \mathbf{e})$ due to the skew-symmetry of $\boldsymbol{S}$. This ensures passivity as $\dot{E} = -(P_{\mathbf{diss}} + P_{\mathbf{ext}}) \leq -P_{\mathbf{ext}}$ makes the internal energy non-increasing when external sources are switched off ($P_{\mathbf{ext}} = 0$).

[1] Variables $(\bar{\mathbf{u}}, \bar{\mathbf{y}}) := \pm(\mathbf{u}, -\mathbf{y})$ are sometimes a preferred convention so that their product ($= -P_{\mathbf{ext}}$) reports power supplied by the outside (see also remark 1).

Example 1 (damped mechanical oscillator)**.** *The oscillator governed by* $m\ddot{z}+r\dot{z}+kz=F_{\text{ext}}$ *can be formulated as*

- 4 separate components*: (i) a mass* [m] *of momentum* π*, a spring* [sp] *of elongation* ξ*; (ii) a damper* [dp] *of velocity* V_{dp}*; (iii) an external actuator* [ext] *applying a force* F_{ext}*:*

	state	*energy* H_n	*flow* $\mathbf{f}$	*effort* $\mathbf{e}$
[m]	$x_1 := \pi$	$\pi^2/(2m)$	$\dot{x}_1 = \dot{\pi}$	$H_1'(x_1) = x_1/m$
[sp]	$x_2 := \xi$	$k\xi^2/2$	$\dot{x}_2 = \dot{\xi}$	$H_2'(x_2) = kx_2$
[dp]	*blue* : *force*		$w := V_{\text{dp}}$	$z(w) := rw$
[ext]	*red* : *velocity*		$y := V_{\text{ext}}$	$u := -F_{\text{ext}}$

- assembled with rigid connections*, meaning that internal* [forces are balanced] *(*$F_{\text{m}}+F_{\text{sp}}+F_{\text{dp}}+(-F_{\text{ext}})=0$*) and all* [velocities are equal] *(*$V_{\text{m}}=V_{\text{sp}}=V_{\text{dp}}=V_{\text{ext}}$*), leading to*

$$\underbrace{\begin{array}{|l|l|}\hline \dot{\pi}=F_{\text{m}} & \dot{x}_1 \\ \dot{\xi}=V_{\text{sp}} & \dot{x}_2 \\ \hline V_{\text{dp}} & w \\ \hline V_{\text{ext}} & y \\ \hline \end{array}}_{\mathbf{f}} = \underbrace{\left[\begin{array}{c|ccc} 0 & -1 & -1 & -1 \\ \hline 1 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 \end{array}\right]}_{\boldsymbol{S}=-\boldsymbol{S}^{\mathsf{T}}} \underbrace{\begin{array}{|l|l|}\hline H_1'(x_1) & V_{\text{m}} = \pi/m \\ H_2'(x_2) & F_{\text{sp}} = k\xi \\ z(w) & F_{\text{dp}} = rV_{\text{dp}} \\ \hline u & -F_{\text{ext}} \\ \hline \end{array}}_{\mathbf{e}}$$

This matches (1) with energy function $H(\boldsymbol{x}) = H_1(x_1)+H_2(x_2)$ *and restores the usual ODE with*[2] $z := \xi$*.*

In this example, nonlinear passive oscillators can be obtained by replacing component laws (see H_n' or z in the table above) satisfying (i-ii). For instance, $H_2(\ell) = \frac{k_1\xi^2}{2} + \frac{k_3\xi^4}{4}$ with $k_1, k_3 > 0$ restores a stiffening Duffing oscillator[3].

Remark 1 (receiver convention)**.** *The flows* $\mathbf{f}$ *and efforts* $\mathbf{e}$ *in (1) denote those* received *by components, whatever their type (i: storing, ii: memoryless, iii: exterior). In physics, this convention is classical for (i-ii), but not systematic for (iii). Note that in example 1, the quantities (iii) are actually those (velocity* V_{ext} *and force* $-F_{\text{ext}}$*)* received *by the external actuator.*

[2] Through the connections, the velocities coincide ($V_{\text{sp}} = V_{\text{m}} = V_{\text{dp}} = V_{\text{ext}} = \dot{z}$) and through the component laws, it comes $z = \xi$ and $m\ddot{z} = \dot{\pi}$.

[3] See Ref. [54] for a passive softening case.

2.2 Differential formulation (DF)

A largely-used formulation is the state-space representation

$$\begin{array}{rrcl} \dot{\mathbf{x}} = & \mathbf{f}(\mathbf{x},\mathbf{u}) & := & \left(\boldsymbol{J}_{\mathbf{xx}} - \boldsymbol{R}_{\mathbf{xx}}\right) \quad \nabla H(\mathbf{x}) + \boldsymbol{G}\mathbf{u}, \\ \mathbf{y} = & \boldsymbol{g}(\mathbf{x},\mathbf{u}) & := & -\boldsymbol{G}^{\intercal} \quad \nabla H(\mathbf{x}), \end{array} \tag{3}$$

where matrices $\boldsymbol{J}_{\mathbf{xx}} = -\boldsymbol{J}_{\mathbf{xx}}^{\intercal}$ (skew-symmetric), $\boldsymbol{R}_{\mathbf{xx}} = \boldsymbol{R}_{\mathbf{xx}}^{\intercal} \succeq 0$ (non-negative symmetric) and $\boldsymbol{G}$ can depend on $\mathbf{x},\mathbf{u}$. This is a particular case of the differential state-space representation

$$\underbrace{\begin{bmatrix}\dot{\mathbf{x}}\\ \mathbf{y}\end{bmatrix}}_{=:\,\mathbf{f}_{\mathrm{D}}} = \left(\underbrace{\begin{bmatrix}\boldsymbol{J}_{\mathbf{xx}} & \boldsymbol{J}_{\mathbf{xu}} \\ * & \boldsymbol{J}_{\mathbf{yu}}\end{bmatrix}}_{=:\,\boldsymbol{J}_{\mathrm{D}} = -\boldsymbol{J}_{\mathrm{D}}^{\intercal}} - \underbrace{\begin{bmatrix}\boldsymbol{R}_{\mathbf{xx}} & \boldsymbol{R}_{\mathbf{xu}} \\ * & \boldsymbol{R}_{\mathbf{yu}}\end{bmatrix}}_{=:\,\boldsymbol{R}_{\mathrm{D}} = \boldsymbol{R}_{\mathrm{D}}^{\intercal} \succeq 0} \right) \underbrace{\begin{bmatrix}\nabla H(\mathbf{x})\\ \mathbf{u}\end{bmatrix}}_{=:\,\mathbf{e}_{\mathrm{D}}\ \text{(D for Differential formulation)}}. \tag{4}$$

Remark 2. *The case (3) corresponds to matrices*

$$\boldsymbol{J}_{\mathrm{D}} = \left[\begin{array}{c|c} \boldsymbol{J}_{\mathbf{xx}} & \boldsymbol{J}_{\mathbf{xu}} = \boldsymbol{G} \\ \hline * & \boldsymbol{J}_{\mathbf{yu}} = \mathbf{0} \end{array}\right] \quad \textit{and} \quad \boldsymbol{R}_{\mathrm{D}} = \left[\begin{array}{c|c} \boldsymbol{R}_{\mathbf{xx}} & \boldsymbol{R}_{\mathbf{xu}} = \mathbf{0} \\ \hline * & \boldsymbol{R}_{\mathbf{yu}} = \mathbf{0} \end{array}\right],$$

involving no feedforward ($\boldsymbol{J}_{\mathbf{yu}} = \boldsymbol{R}_{\mathbf{yu}} = \mathbf{0}$) and a conservative routing between ports and energy-storing components ($\boldsymbol{R}_{\mathbf{xu}} = \mathbf{0}$).

Only the flows and efforts of (i) and (iii) are represented, whereas laws of (ii) are converted into connections through matrices $\boldsymbol{J}$ (conservative part) and $\boldsymbol{R}$ (dissipative part), as

$$P_{\mathbf{stored}} + P_{\mathbf{ext}} = \mathbf{e}_{\mathrm{D}}^{\intercal}\,\mathbf{f}_{\mathrm{D}} = \underbrace{\mathbf{e}_{\mathrm{D}}^{\intercal}\,\boldsymbol{J}\,\mathbf{e}_{\mathrm{D}}}_{=0} - \underbrace{\mathbf{e}_{\mathrm{D}}^{\intercal}\,\boldsymbol{R}\,\mathbf{e}_{\mathrm{D}}}_{\geq 0}. \tag{5}$$

The dissipated power due to (ii) reformulates as $P_{\mathbf{diss},\mathrm{D}} := \mathbf{e}_{\mathrm{D}}^{\intercal}\boldsymbol{R}\,\mathbf{e}_{\mathrm{D}} \geq 0$. It reduces to $\nabla H(\mathbf{x})^{\intercal}\boldsymbol{R}_{\mathbf{xx}}\nabla H(\mathbf{x})$ for equation (3).

Example 2. *The oscillator corresponds to such a case with*

$$\underbrace{\begin{array}{c|c} F_{\mathrm{m}} & \dot{x}_1 \\ V_{\mathrm{sp}} & \dot{x}_2 \\ \hline V_{\mathrm{ext}} & y \end{array}}_{\mathbf{f}_{\mathrm{D}}} = \left(\underbrace{\left[\begin{array}{ccc|c} 0 & -1 & -1 & \\ 1 & 0 & 0 & \\ \hline 1 & 0 & 0 & \end{array}\right]}_{\boldsymbol{J} = -\boldsymbol{J}^{\intercal}} - \underbrace{\left[\begin{array}{ccc|c} r & 0 & 0 & \\ 0 & 0 & 0 & \\ \hline 0 & 0 & 0 & \end{array}\right]}_{\boldsymbol{R} = \boldsymbol{R}^{\intercal} \succeq 0} \right) \underbrace{\begin{array}{c|c} H_1'(x_1) & V_{\mathrm{m}} \\ H_2'(x_2) & F_{\mathrm{sp}} \\ \hline u & -F_{\mathrm{ext}} \end{array}}_{\mathbf{e}_{\mathrm{D}}}$$

so that $\boldsymbol{J}_{\mathbf{xx}} = \begin{bmatrix}0 & -1\\ 1 & 0\end{bmatrix}$, $\boldsymbol{R}_{\mathbf{xx}} = \begin{bmatrix}r & 0\\ 0 & 0\end{bmatrix}$ *and* $\boldsymbol{G} = \begin{bmatrix}-1\\ 0\end{bmatrix}$ *in (3). Note that contrary to* $\boldsymbol{J}$ *(or* $\boldsymbol{S}$ *in example 1), matrix* $\boldsymbol{R}$ *is not independent of all the component laws: it mixes the routing ($V_{\mathrm{m}} \to V_{\mathrm{dp}} \to w$) and the reconstruction of the damping force ($z(w) = rw$) to produce rV_{m} in the force balance equation.*

2.3 Relations between these formulations

A differential formulation (4) simply stems from formulation (1), under the following conditions (met in example 1):

$$\mathbf{z}(\mathbf{w}) = \boldsymbol{\Gamma}(\mathbf{w})\,\mathbf{w} \;\text{ with } \boldsymbol{\Gamma}+\boldsymbol{\Gamma}^\intercal \succeq 0, \tag{6a}$$

$$\boldsymbol{S}_{\mathbf{ww}} = \mathbf{0} \;\text{ and }\; \boldsymbol{P} := [-\boldsymbol{S}_{\mathbf{xw}}^\intercal, \boldsymbol{S}_{\mathbf{wu}}] \text{ is independent of } \mathbf{w}. \tag{6b}$$

In this case, the algebraic part of (1) admits a unique solution

$$\mathbf{w} = \boldsymbol{W}(\mathbf{e}_\mathrm{D}) := \boldsymbol{P}\,\mathbf{e}_\mathrm{D}. \tag{7}$$

Expressing the efforts of memoryless components through the function composition[4] $\mathbf{z}\circ\boldsymbol{W}$ leads to (4) with

$$\boldsymbol{J}_\mathrm{D} = \begin{bmatrix} \boldsymbol{S}_{\mathbf{xx}} & \boldsymbol{S}_{\mathbf{xu}} \\ * & \boldsymbol{S}_{\mathbf{yu}} \end{bmatrix} - \boldsymbol{P}^\intercal \boldsymbol{J}_\Gamma \boldsymbol{P} \qquad \text{with } \boldsymbol{J}_\Gamma := \frac{1}{2}(\boldsymbol{\Gamma}_\mathrm{D} - \boldsymbol{\Gamma}_\mathrm{D}^\intercal), \tag{8a}$$

$$\boldsymbol{R}_\mathrm{D} = \boldsymbol{P}^\intercal \boldsymbol{R}_\Gamma \boldsymbol{P} \succeq 0 \qquad \text{with } \boldsymbol{R}_\Gamma := \frac{1}{2}(\boldsymbol{\Gamma}_\mathrm{D} + \boldsymbol{\Gamma}_\mathrm{D}^\intercal), \tag{8b}$$

and where $\boldsymbol{\Gamma}_\mathrm{D} = \boldsymbol{\Gamma}\circ\boldsymbol{W}$ can depend on[5] $\mathbf{e}_\mathrm{D}$.

Example 3. *In example 1, $\boldsymbol{\Gamma} = r$ ($\boldsymbol{J}_\Gamma = 0$ and $\boldsymbol{R}_\Gamma = r$) and $\boldsymbol{P} = [1\;0|0]$ yield the result stated in example 2.*

▷You are now ready for exercice 1 (questions Q1 and Q2, see § 5).

2.4 Hybrid formulation (HF)

A reformulation can also be derived by eliminating a part A of components ($\mathbf{w}_A \mapsto \mathbf{z}_A(\mathbf{w}_A, \mathbf{w}_B)$) for which (6a-6b) are met, while the complementary part B ($\mathbf{w}_B \mapsto \mathbf{z}_B(\mathbf{w}_A, \mathbf{w}_B)$) is left untreated. Renaming $\mathbf{w}_B$ and $\mathbf{z}_B$ without label B for sake of conciseness, this case leads to the *hybrid formulation*

$$\underbrace{\begin{bmatrix} \dot{\mathbf{x}} \\ \mathbf{w} \\ \mathbf{y} \end{bmatrix}}_{=:\,\mathbf{f}_\mathrm{h}} = \Bigg(\underbrace{\begin{bmatrix} \boldsymbol{J}_{\mathbf{xx}} & \boldsymbol{J}_{\mathbf{xw}} & \boldsymbol{J}_{\mathbf{xu}} \\ * & \boldsymbol{J}_{\mathbf{ww}} & \boldsymbol{J}_{\mathbf{wu}} \\ * & * & \boldsymbol{J}_{\mathbf{yu}} \end{bmatrix}}_{=:\,\boldsymbol{J}_\mathrm{h} = -\boldsymbol{J}_\mathrm{h}^\intercal} - \underbrace{\begin{bmatrix} \boldsymbol{R}_{\mathbf{xx}} & \boldsymbol{R}_{\mathbf{xw}} & \boldsymbol{R}_{\mathbf{xu}} \\ * & \boldsymbol{R}_{\mathbf{ww}} & \boldsymbol{R}_{\mathbf{wu}} \\ * & * & \boldsymbol{R}_{\mathbf{yu}} \end{bmatrix}}_{=:\,\boldsymbol{R}_\mathrm{h} = \boldsymbol{R}_\mathrm{h}^\intercal \succeq 0} \Bigg) \underbrace{\begin{bmatrix} \nabla H(\mathbf{x}) \\ \mathbf{z}(\mathbf{w}) \\ \mathbf{u} \end{bmatrix}}_{=:\,\mathbf{e}_\mathrm{h}\ \text{(h for hybrid)}}, \tag{9}$$

that is, $\mathbf{f}_\mathrm{h} = \boldsymbol{M}\,\mathbf{e}_\mathrm{h}$ with $\boldsymbol{M} := \boldsymbol{J}_\mathrm{h} - \boldsymbol{R}_\mathrm{h}$, for which the dissipated power corresponds to

$$P_{\mathbf{diss},\mathrm{h}} = \underbrace{\mathbf{z}(\mathbf{w})^\intercal \mathbf{w}}_{\geq 0} + \underbrace{\mathbf{e}_\mathrm{h}^\intercal \boldsymbol{R}_\mathrm{h}\,\mathbf{e}_\mathrm{h}}_{\geq 0} \geq 0. \tag{10}$$

Remark 3 (Link with formulation (1))**.** *Formally, this hybrid formulation (9) can be viewed as (1), in which $\boldsymbol{S} \equiv \boldsymbol{J}_\mathrm{h} - \boldsymbol{R}_\mathrm{h}$ is no longer skew-symmetric but includes a non-positive symmetric part ($-\boldsymbol{R}_\mathrm{h}$) partly responsible for dissipation.*

[4] As in example 2, this mixes a routing $\boldsymbol{P}$ and a component law $\mathbf{z}$.

[5] defining a PHS under an integrability condition on $\boldsymbol{J}_\mathrm{D}$ [4, p.48] or a so-called pseudo-PHS otherwise.

2.5 Interconnection of several PHS

Consider a collection of N independent systems $\mathscr{S}_{n=1,\dots,N}$, all described by (1). Denote their gathered variables

$$\mathbf{v}^\intercal := [\mathbf{v}_{\mathscr{S}_1}^\intercal, \dots, \mathbf{v}_{\mathscr{S}_N}^\intercal]^\intercal, \text{ for all labels } \mathbf{v} = \mathbf{x},\ \mathbf{w},\ \mathbf{u} \text{ or } \mathbf{y},$$

and the associated block-diagonal matrices

$$\boldsymbol{S}_{\mathbf{v}'\mathbf{v}} = \operatorname{diag}(\boldsymbol{S}_{\mathbf{v}'_{\mathscr{S}_1}\mathbf{v}_{\mathscr{S}_1}}, \dots, \boldsymbol{S}_{\mathbf{v}'_{\mathscr{S}_N}\mathbf{v}_{\mathscr{S}_N}}) \text{ for all labels } \mathbf{v}' \text{ and } \mathbf{v},$$

to form the description (1) of the global PHS $\mathscr{S}$.

Now, interconnect these systems according to a causal[6] memoryless passive[7] law

$$\mathbf{u}_c = \mathbf{z}_c(\mathbf{y}_c), \quad \text{(interconnection law)} \tag{11}$$

between some ports of $\mathscr{S} \equiv (\mathscr{S}_1, \dots, \mathscr{S}_N)$, selected as

$$[\mathbf{y}_c, \mathbf{u}_c] = \boldsymbol{C}^\intercal[\mathbf{y}, \mathbf{u}], \qquad \text{(connected ports)} \tag{12a}$$
$$[\mathbf{y}_f, \mathbf{u}_f] = \boldsymbol{F}^\intercal[\mathbf{y}, \mathbf{u}], \qquad \text{(free ports)} \tag{12b}$$

where the concatenation of selection matrices $[\boldsymbol{C}^T, \boldsymbol{F}^\intercal]$ defines a permutation matrix that rearranges the ports.

Then, the interconnected global system is a PHS governed by

$$\left[\begin{array}{c} \mathbf{x} \\ \mathbf{w} \\ \hdashline \mathbf{y}_c \\ \hline \mathbf{y}_f \end{array}\right] = \underbrace{\left[\begin{array}{ccc:c|c} \boldsymbol{S}_{\mathbf{xx}} & \boldsymbol{S}_{\mathbf{xw}} & \boldsymbol{S}_{\mathbf{xu}}\boldsymbol{C} & \boldsymbol{S}_{\mathbf{xu}}\boldsymbol{F} \\ * & \boldsymbol{S}_{\mathbf{ww}} & \boldsymbol{S}_{\mathbf{wu}}\boldsymbol{C} & \boldsymbol{S}_{\mathbf{wu}}\boldsymbol{F} \\ \hdashline * & * & \boldsymbol{C}^\intercal\boldsymbol{S}_{\mathbf{yu}}\boldsymbol{C} & \boldsymbol{C}^\intercal\boldsymbol{S}_{\mathbf{yu}}\boldsymbol{F} \\ \hline * & * & * & \boldsymbol{F}^\intercal\boldsymbol{S}_{\mathbf{yu}}\boldsymbol{F} \end{array}\right]}_{\text{skew-symmetric}} \left[\begin{array}{c} \nabla H(\mathbf{x}) \\ \mathbf{z}(\mathbf{w}) \\ \hdashline \mathbf{z}_c(\mathbf{y}_c) \\ \hline \mathbf{u}_f \end{array}\right]. \tag{13}$$

Under condition (6a-6b) adapted to[8] $\mathbf{z}_c$, the connection variable $\mathbf{y}_c$ may be fully (otherwise, partially) reduced following the processes described in sections 2.3-2.4. In this case, $\mathbf{z}_c$ defines a fully (otherwise, partially) causal interconnection. Moreover, in accordance with remark 3, formulation (13) can be adapted to the hybrid formulation.

Remark 4 (PyPHS)**.** *In [33], a method which allows the automatic derivation of a complete model from components and interconnections is described for electronic circuits. A corresponding Python library PyPHS [27, 77, 5] is available for multi-physical systems. It automates the derivation of the PHS, of the discrete gradient-based numerical scheme (presented below) with a Newton-Raphson solver, and it generates C++ code for simulation. It is aimed at numerical simulation, such as* 20-sim *based on bond graphs (see also the recent work [9, 10]), Modelica/Dymola. See also work on dedicated languages [11].*

[6] law feeding inputs of $\mathscr{S}$ from observed outputs of $\mathscr{S}$.

[7] as in § 2.1 (ii)

[8] The condition is $\boldsymbol{C}^\intercal\boldsymbol{S}_{\mathbf{yu}}\boldsymbol{C} = 0$ and that $\boldsymbol{P} = \boldsymbol{C}^\intercal\left[-\boldsymbol{S}_{\mathbf{xu}}, -\boldsymbol{S}_{\mathbf{wu}}, +\boldsymbol{S}_{\mathbf{yu}}\boldsymbol{F}\right]$ does not depend on $\mathbf{w}_c$.

2.6 Elementary non-causal interconnections

Some systems involve non-causal interconnections (no relation $\mathbf{f} = \boldsymbol{S}\mathbf{e}$ is available). Examples[9] are components that share the same effort (constraints of type $[\mathbf{e}]_i = [\mathbf{e}]_j$), or, when not caused by the complementary part of the system, the same flow ($[\mathbf{f}]_i = [\mathbf{f}]_j$). The general case can be addressed by considering kernel representations of Dirac structures[10] (see e.g. [2]) or specific differential-algebraic formulations (see e.g. [12, 13]).

Two types of elementary (but common) interconnections of energy-storing components are detailed below, that lead to an equivalent component (single Hamiltonian).

2.6.1 Shared efforts

Hamiltonians H_1 and H_2 are supposed to be $\mathscr{C}^1$ convex non-negative functions, so that their gradients are continuous monotone. The two components share the same effort

$$\mathbf{e} := \mathbf{E}_1(\mathbf{x}_1) = \mathbf{E}_2(\mathbf{x}_2), \text{ with } \mathbf{E}_i := \nabla H_i \text{ of inverse } \mathbf{X}_i := \mathbf{E}_i^{-1} : \mathbf{e} \longmapsto \mathbf{x}_i. \tag{14}$$

The set of these components receives the total flow $\mathbf{f} = \dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2$. This extensivity property on the flow makes $\mathbf{f}$ interpretable as the time derivative of a total state

$$\mathbf{x} := \mathbf{x}_1 + \mathbf{x}_2 = \mathbf{X}(\mathbf{e}), \text{ with } \mathbf{X} := \mathbf{X}_1 + \mathbf{X}_2 \text{ monotone of inverse } \mathbf{E} := \mathbf{X}^{-1} : \mathbf{x} \mapsto \mathbf{e}. \tag{15}$$

Finally, expressing the total energy $H_1(\mathbf{x}_1) + H_2(\mathbf{x}_2)$ as a function of the total state $\mathbf{x}$ provides the Hamiltonian of the equivalent component

$$H : \mathbf{x} \in \mathbb{R}^N \longmapsto [H_1 \circ \mathbf{X}_1 + H_2 \circ \mathbf{X}_2] \circ \mathbf{E}(\mathbf{x}). \tag{16}$$

Note that H is convex (sum of convex functions), continuous (functions in (16) are all continuous) and such that $H_1(\mathbf{0}) = H_2(\mathbf{0}) = 0 \Rightarrow H(\mathbf{0}) = 0$. Its $\mathscr{C}^1$-regularity (expected to be inherited from H_1, H_2) is also satisfied (a proof not detailed here, also valid for $\mathscr{C}^k$-cases, is based on the use of push-forward measures).

2.6.2 Shared flows

In the case of energy-storing components that share a flow, the two components with Hamiltonian $H_1(\mathbf{x}_1)$ and $H_2(\mathbf{x}_2)$ receive the same flow $\dot{\mathbf{x}}_1 = \dot{\mathbf{x}}_2$, so that $\mathbf{x}_1(t) - \mathbf{x}_1(0) = \mathbf{x}_2(t) - \mathbf{x}_2(0)$. An equivalent component can then be defined by a Hamiltonian

$$H_{\mathbf{X}_0} : \mathbf{x} \in \mathbb{R}^N \longmapsto H_1(\mathbf{x}) + H_2(\mathbf{x} + \mathbf{X}_0), \text{ with } \mathbf{X}_0 := \mathbf{x}_2(t=0) - \mathbf{x}_1(t=0), \tag{17}$$

which depends on the initial condition through $\mathbf{X}_0$.

Note that hamiltonian H is chosen such that $H_{\mathbf{X}_0}(\mathbf{0}) = 0$. It admits a lower bound, possibly negative. Another choice for defining its non-negative version is $H^+_{\mathbf{X}_0}(\mathbf{x}) := H_{\mathbf{X}_0}(\mathbf{x}) - \min_{\mathbf{x}} H_{\mathbf{X}_0}(\mathbf{x})$. For quadratic hamilonians, these two choices coincide.

Note also that, as mentioned above, when the flow $\mathbf{f}$ is caused by the complementary part of the system, this interconnection is not to be processed as a constraint (writing $\dot{\mathbf{x}}_1 = \mathbf{f}, \dot{\mathbf{x}}_2 = \mathbf{f}$), but (17) can still be used for order reduction.

[9] Shared flow: capacitors in series, coils in parallel, cascaded springs, etc. Shared effort: capacitors in parallel, coils in series, springs attached together at both ends, etc.

[10] Efforts and flows are related as $\boldsymbol{K}\mathbf{f} + \boldsymbol{L}\mathbf{e} = \mathbf{0}$ where $\boldsymbol{K}, \boldsymbol{L}$ are $N \times N$ interconnection matrices, which satisfy $\boldsymbol{K}\boldsymbol{L}^\intercal + \boldsymbol{L}\boldsymbol{K}^\intercal = \mathbf{0}$ and $\text{rank}[\boldsymbol{K}\ \boldsymbol{L}] = N$. Matrix $\boldsymbol{S} = -\boldsymbol{K}^{-1}\boldsymbol{L}$ exists if $\boldsymbol{K}$ is invertible.

2.7 PHS shifting

When the time-varying variables of a system have a bias[11] (non-centered signals), it can be beneficial, for e.g. analysis or numerical issues, to formulate the exact equations that govern their centered fluctuating parts. The shifting operation is the transformation that produces these equations from the formulation governing the original variables.

Consider a PHS described by the hybrid formulation (9) for sake of generality[12]. Assume that $\boldsymbol{M} := \boldsymbol{J}_\mathrm{h} - \boldsymbol{R}_\mathrm{h}$ is constant. Decompose all the original variables ($\mathrm{var}(t)$) into the sum of a bias value ($\mathrm{var}^\star$) and a fluctuation ($\widetilde{\mathrm{var}}(t)$). Variables $\mathrm{var}^\star$ correspond to equilibrium values, as they coincides with the original variables when the system is at rest (no fluctuation). These variables are related as:

$$\text{(PHS)}: \underbrace{\begin{bmatrix}\dot{\mathbf{x}}\\ \mathbf{w}\\ \mathbf{y}\end{bmatrix}}_{\mathbf{f}_\mathrm{h}(t)} = \boldsymbol{M} \underbrace{\begin{bmatrix}\nabla H(\mathbf{x})\\ \mathbf{z}(\mathbf{w})\\ \mathbf{u}\end{bmatrix}}_{\mathbf{e}_\mathrm{h}(t)} \quad \text{and} \quad (\text{PHS})^\star: \underbrace{\begin{bmatrix}\dot{\mathbf{x}}^\star = \mathbf{0}\\ \mathbf{w}^\star\\ \mathbf{y}^\star\end{bmatrix}}_{\mathbf{f}_\mathrm{h}^\star} = \boldsymbol{M} \underbrace{\begin{bmatrix}\nabla H(\mathbf{x}^\star)\\ \mathbf{z}(\mathbf{w}^\star)\\ \mathbf{u}^\star\end{bmatrix}}_{\mathbf{e}_\mathrm{h}^\star}. \tag{18}$$

The shifted system $\widetilde{(\text{PHS})} \equiv$ (PHS)-$(\text{PHS})^\star$ reads $\widetilde{\mathbf{f}}_\mathrm{h} = \boldsymbol{M}\widetilde{\mathbf{e}}_\mathrm{h} \;\equiv\; \mathbf{f}_\mathrm{h}(t) - \mathbf{f}_\mathrm{h}^\star = \boldsymbol{M}(\mathbf{e}_\mathrm{h}(t) - \mathbf{e}_\mathrm{h}^\star)$. Writing efforts w.r.t. $\mathrm{var}^\star$ and $\widetilde{\mathrm{var}}$ leads to $\nabla H(\widetilde{\mathbf{x}} + \mathbf{x}^\star) - \nabla H(\mathbf{x}^\star)$ for energy-storing components (i), $\mathbf{z}(\widetilde{\mathbf{w}} + \mathbf{w}^\star) - \mathbf{z}(\mathbf{w}^\star)$ for memoryless components (ii), and $\widetilde{\mathbf{u}}$ for ports (iii). $\widetilde{(\text{PHS})}$ must involve functions $\mathbf{x}^\star \mapsto \widetilde{H_{\mathbf{x}^\star}}(\widetilde{\mathbf{x}})$ and $\mathbf{w}^\star \mapsto \widetilde{\mathbf{z}_{\mathbf{w}^\star}}(\widetilde{\mathbf{w}})$, in which $\mathbf{x}^\star$ and $\mathbf{w}^\star$ are considered as parameters. Integrating expression (i) from $\mathbf{0}$ w.r.t. $\widetilde{\mathbf{x}}$ defines $\widetilde{H_{\mathbf{x}^\star}}(\widetilde{\mathbf{x}})$, leading to

$$\widetilde{(\text{PHS})}: \underbrace{\begin{bmatrix}\dot{\widetilde{\mathbf{x}}}\\ \widetilde{\mathbf{w}}\\ \widetilde{\mathbf{y}}\end{bmatrix}}_{\widetilde{\mathbf{f}}_\mathrm{h}(t)} = \boldsymbol{M} \underbrace{\begin{bmatrix}\nabla \widetilde{H_{\mathbf{x}^\star}}(\widetilde{\mathbf{x}})\\ \widetilde{\mathbf{z}_{\mathbf{w}^\star}}(\widetilde{\mathbf{w}})\\ \widetilde{\mathbf{u}}\end{bmatrix}}_{\widetilde{\mathbf{e}}_\mathrm{h}(t)} \quad \text{with} \quad \begin{cases} \widetilde{H_{\mathbf{x}^\star}}(\widetilde{\mathbf{x}}) & := H(\widetilde{\mathbf{x}} + \mathbf{x}^\star) - \nabla H(\mathbf{x}^\star)^\intercal \widetilde{\mathbf{x}} - H(\mathbf{x}^\star), \\ \widetilde{\mathbf{z}_{\mathbf{w}^\star}}(\widetilde{\mathbf{x}}) & := \mathbf{z}(\widetilde{\mathbf{w}} + \mathbf{w}^\star) - \mathbf{z}(\mathbf{w}^\star). \end{cases} \tag{19}$$

Remark 5 (Passivity)**.** *The shifted system $\widetilde{(\text{PHS})}$ is passive w.r.t. input $\widetilde{\mathbf{u}}$ and output $\widetilde{\mathbf{y}}$, if* $\widetilde{\mathbf{z}_{\mathbf{w}^\star}}(\widetilde{\mathbf{w}})^\intercal \widetilde{\mathbf{w}} \geq 0$.

Remark 6 (Bregman distance)**.** *As noticed in [14, Rk 6.5.2, p. 137], the shifted Hamiltonian can be interpreted as*

$$\widetilde{H_{\mathbf{x}^\star}}(\widetilde{\mathbf{x}}) := D_H(\mathbf{x}^\star + \widetilde{\mathbf{x}}, \mathbf{x}^\star) \text{ with } D_H(\mathbf{x}, \mathbf{x}^\star) := H(\mathbf{x}) - H(\mathbf{x}^\star) - \nabla H(\mathbf{x}^\star)^\intercal (\mathbf{x} - \mathbf{x}^\star), \tag{20}$$

where D_H defines the Bregman distance for convex functions H (see also [15, 16]).

[11] This occurs in electronics (operating point configuration through a power supply), electromagnetism (non-zero average charge, or magnets), in mechanics (pre-stress), in fluid mechanics, thermodynamics and acoustics (average density, velocity, temperature/entropy or atmospheric pressure), etc.

[12] in the generic sense that $\mathbf{w}$ can be empty or $\boldsymbol{R}_\mathrm{h}$ can be zero.

3 Discrete gradient method

3.1 Objective and principle

This section presents a numerical method to compute discrete-time trajectories of PHS, which preserves passivity, that is, for which (2) is satisfied[13] on a time grid $t \in (\delta t)\mathbb{Z}$, and in which $P_{\textbf{stored}}$ still corresponds to a time-variation of energy E in discrete-time with *consistent sign (including zeroness for conservative systems)*. Such an adequate and simple discrete-time approximation of the energy variation is

$$P_{stored}(t) \approx [\, E(t+\delta t) - E(t) \,] \,/\, \delta t, \tag{21}$$

so that the energy still increases or decreases according to the sign of $-(P_{\textbf{diss}}+P_{\textbf{ext}})$. Introducing the *discrete derivation operator* D as

$$\forall \boldsymbol{f} \in \mathscr{C}^1(\mathbb{R},\mathbb{R}^N), \quad D\boldsymbol{f}(s,\delta) := \begin{cases} \frac{\boldsymbol{f}(s+\delta)-\boldsymbol{f}(s)}{\delta} & \text{if } \delta \neq 0, \\ \boldsymbol{f}'(s) & \text{otherwise,} \end{cases} \tag{22}$$

this choice consists in considering the discrete variation $DE(t,\delta t)$ instead of its continuous limit case $DE(t,0)$. The objective is to propose a numerical method to simulate the trajectories governed by (1, 4 or 9) and that satisfies this principle.

3.2 Case I: mono-variate Hamiltonian

For a mono-variate Hamiltonian $H : x \in \mathbb{R} \mapsto \mathbb{R}_+$, the time-variation $DE(t,\delta t)$ of energy $E(t) := H \circ x(t)$ is given by

$$\begin{aligned} D[H\circ x](t,\delta t) &= \frac{H(x(t+\delta t)) - H(x(t))}{\delta t} \\ &= DH\Big(x(t), x(t+\delta t)-x(t)\Big)\,\frac{x(t+\delta t)-x(t)}{\delta t} \\ &= DH\Big(x(t), Dx(t,\delta t)\,\delta t\Big)\,Dx(t,\delta t). \end{aligned} \tag{23}$$

This formula generalises the chain rule for derivatives ($\dot{E} = H'(x(t))\,\dot{x}(t)$) to operator D. Therefore, jointly replacing $\dot{x}(t)$ and $\nabla H(x(t))$ in (1) by their approximations $Dx(t,\delta t)$ and $DH\Big(x(t), Dx(t,\delta t)\,\delta t\Big)$ defines a family (continuously parameterised by $\delta t > 0$) of discrete-time PHS with the expected power balance.

A lighter writing is obtained by denoting $E[k]$, $x[k]$ (etc) the signals sampled at time $t_k := k\,\delta t$ ($k \in \mathbb{Z}$ or, given an initial condition $\mathbf{x}[0]$, $k \in \mathbb{N}$), and $\delta x[k]$ the increment $x[k+1]-x[k] = Dx(t_k,\delta t_k)\delta t$. The substitutions to operate in (1, 4 or 9) then rewrite

$$(\mathrm{I}) \begin{cases} \dot{x}(t) & \to & \delta x[k]/\delta t \\ \nabla H(x(t)) & \to & DH(x[k],\delta x[k]) \end{cases} \text{ with } \delta x[k] := x[k+1]-x[k]. \tag{24}$$

[13] and by extension (5) and (10)

3.3 Case II: Hamiltonian with separated variables

Assuming that $H(\mathbf{x}=[x_1,\dots,x_N]^\intercal) \overset{(\text{II})}{:=} \sum_{n=1}^N H_n(x_n)$, equation (23) becomes

$$D[H\circ\mathbf{x}](t,\delta t) \overset{(22)}{=} \sum_{n=1}^N \frac{H_n(x_n(t+\delta t)) - H_n(x_n(t))}{\delta t}$$
$$\overset{(23)}{=} \sum_{n=1}^N DH_n\Big(x_n(t), Dx_n(t,\delta t)\,\delta t\Big)\, Dx_n(t,\delta t), \qquad (25)$$

$$\text{rewritten as } D[H\circ\mathbf{x}](t,\delta t) = \nabla_D H\Big(\mathbf{x}(t), D\mathbf{x}(t,\delta t)\,\delta t\Big)^\intercal\, D\mathbf{x}(t,\delta t), \qquad (26)$$

and the substitutions (24) become vectorial

$$(\text{II or III})\begin{cases} \dot{\mathbf{x}}(t) & \to & \delta\mathbf{x}[k]/\delta t \\ \nabla H(x(t)) & \to & \nabla_D H(\mathbf{x}[k],\delta\mathbf{x}[k]) \text{ with } \delta\mathbf{x}[k] := \mathbf{x}[k+1]-\mathbf{x}[k], \end{cases} \qquad (27)$$

where operator $\nabla_D H$ can be defined from (25-26) by $\nabla_D H = \nabla_{\text{II}} H$ with

$$\nabla_{\text{II}} H(\mathbf{x},\delta\mathbf{x}) := \begin{bmatrix} DH_1(x_1,\delta x_1) \\ \vdots \\ DH_N(x_N,\delta x_N) \end{bmatrix} \text{ under assumption (II).} \qquad (28)$$

In general, any operator ∇_D satisfying (26) is called a *discrete gradient*.

▷ Exercise 1 (Q3a)

3.4 Case III: general case

In the case III for which $H(\mathbf{x}=[x_1,\dots,x_N]^\intercal)$ is any multivariate function (II is no longer assumed), equation (26) remains valid for the discrete gradient

$$\nabla_{\text{III}} H(\mathbf{x},\delta\mathbf{x}) := \begin{bmatrix} D[\sigma_1 H](x_1,\delta x_1) \\ \vdots \\ D[\sigma_N H](x_N,\delta x_N) \end{bmatrix} \text{ with } \sigma_n H : \xi \mapsto H\begin{pmatrix} x_1 \\ \vdots \\ x_{n-1} \\ \xi \\ x_{n+1}+\delta x_{n+1} \\ \vdots \\ x_N+\delta x_N \end{pmatrix}. \qquad (29)$$

Indeed, since $[\sigma_n H](x_n+\delta x_n) \overset{(*)}{=} [\sigma_{n-1}H](x_{n-1})$ for $2\le n\le N$, it follows that

$$\nabla_{\text{III}} H(\mathbf{x},\delta\mathbf{x})^\intercal\,\delta\mathbf{x} = \Big[\, D[\sigma_1 H](x_1,\delta x_1)\ ,\ \dots\ ,\ D[\sigma_N H](x_N,\delta x_N)\,\Big]\,\delta\mathbf{x}$$
$$\overset{(22)}{=} \sum_{n=1}^N \Big(\sigma_n H(x_n+\delta x_n) - \sigma_n H(x_n)\Big)$$
$$= \sigma_1 H(x_1+\delta x_1) + \sum_{n=2}^N \sigma_n H(x_n+\delta x_n) - \sum_{n=1}^{N-1}\sigma_n H(x_n) - \sigma_N H(x_N)$$
$$\overset{(*)}{=} \sigma_1 H(x_1+\delta x_1) - \sigma_N H(x_N) = H(\mathbf{x}+\delta\mathbf{x}) - H(\mathbf{x}), \qquad (30)$$

from which (26) stems. Then, the substitution (27) with $\nabla_D H = \nabla_{\text{III}} H$ provides a power-balanced numerical method.

3.5 Symmetric discrete gradient

For all permutations $\pi \in \mathscr{P}(N)$ (permutations on $\{1,2,\ldots,N\}$), consider the transformation that reorders the state variables and denote for simplicity

$$\pi(\mathbf{x} = [x_1,\ldots,x_N]^\intercal) :\equiv [x_{\pi(1)},\ldots,x_{\pi(N)}]^\intercal \text{ (without ambiguity)}, \tag{31a}$$

$$\text{and } H_\pi := H \circ \pi^{-1}, \tag{31b}$$

so that $H_\pi(\pi(\mathbf{x})) = H(\mathbf{x})$ restores the Hamiltonian for the reordered state variables $\pi(\mathbf{x})$. From (31b), the n-th component of $\nabla H_\pi(\pi(\mathbf{x}))$ is the $\pi^{-1}(n)$-th component of $\nabla H(\mathbf{x})$, that is, $\pi \circ \nabla H_\pi \circ \pi = \nabla H$. Technically, this means that the gradient ∇ is invariant under the transformation

$$\boldsymbol{T}^\pi : \nabla \longmapsto (H \mapsto \pi \circ \nabla H_\pi \circ \pi), \tag{32}$$

that is, $\boldsymbol{T}^\pi(\nabla) = \nabla$. Physically, this means that reordering the state variables does not modify the effort laws, which consistently translates the physical principle of invariance to any arbitrary declaration order of the state variables.

To satisfy this physical principle, a discrete gradient $\nabla_D H(\mathbf{x}, \delta\mathbf{x})$ must satisfy the invariance w.r.t. to the joint reordering of $(\mathbf{x}, \delta\mathbf{x})$, that is, $\boldsymbol{T}_D^\pi(\nabla_D) = \nabla_D$ for the adapted transformation

$$\boldsymbol{T}_D^\pi : \nabla_D \longmapsto (\, H \mapsto \pi \circ [\nabla_D H_\pi](\pi \circ \cdot, \pi \circ \cdot)\,). \tag{33}$$

As the sequence $(\sigma_{\pi(n)})_{n=1,\ldots,N}$ differs from $(\sigma_n)_{n=1,\ldots,N}$, ∇_{III} does not satisfy this invariance: for all $\pi \in \mathscr{P}(N)$, $\boldsymbol{T}_D^\pi(\nabla_{\text{III}})$ defines a discrete gradient[14] denoted ∇_{III}^π below, but $\boldsymbol{T}_D^\pi(\nabla_{\text{III}}) \neq \nabla_{\text{III}}$ if $\pi \neq I_d$.

For all discrete gradients ∇_D that apply to multivariate Hamiltonians, the symmetric discrete gradient $\nabla_D = \overline{\nabla_D}$ defined by

$$\nabla_D \longmapsto \overline{\nabla_D} := \frac{1}{N!} \sum_{\pi \in \mathscr{P}(N)} \boldsymbol{T}_D^\pi(\nabla_D) \quad \text{(symmetric discrete gradient)}, \tag{34}$$

restores the invariance of the original PHS to any arbitrary declaration order of the state variables.

Example 4 (Non-equivalent discrete gradients)**.** *Consider the (dimensionless) Hamiltonian* $H(\mathbf{x} := [x_1, x_2]^\intercal) = \frac{1}{2}x_1^2(1+\varepsilon x_2^2)$ *and denote* π *the swapping application (*$\pi(1) = 2$*,* $\pi(2) = 1$*). Deriving the discrete gradients yields*

$\nabla_{\text{III}} H$	$\nabla_{\text{III}}^\pi H := \boldsymbol{T}_D^\pi \nabla_{\text{III}} H$	$\overline{\nabla_{\text{III}}} H$
$\begin{bmatrix} (x_1 + \frac{\delta x_1}{2})\Big(1 + \varepsilon(x_2 + \delta x_2)^2\Big) \\ \frac{x_1^2}{2}\,\varepsilon\,(2x_2 + \delta x_2) \end{bmatrix}$	$\begin{bmatrix} (x_1 + \frac{\delta x_1}{2})\,(1 + \varepsilon x_2^2) \\ \frac{(x_1+\delta x_1)^2}{2}\,\varepsilon\,(2x_2 + \delta x_2) \end{bmatrix}$	$\begin{bmatrix} (x_1 + \frac{\delta x_1}{2})\Big(1 + \varepsilon\frac{x_2^2 + (x_2+\delta x_2)^2}{2}\Big) \\ \frac{1}{2}\frac{x_1^2 + (x_1+\delta x_1)^2}{2}\,\varepsilon\,(2x_2 + \delta x_2) \end{bmatrix}$

[14] (30) is satisfied replacing $(H, \mathbf{x})$ by $(H_\pi, \pi \circ \mathbf{x})$ and leaving σ_n unchanged.

Remark 7 (Midpoint discrete gradient)**.** *The operator given by*

$$\overline{\nabla_{mid}}H(\mathbf{x},\delta\mathbf{x}) := \begin{cases} \nabla H(\mathbf{x}+\frac{1}{2}\delta\mathbf{x}) + \dfrac{H(\mathbf{x}+\delta\mathbf{x}) - H(\mathbf{x}) - \nabla H(\mathbf{x}+\frac{1}{2}\delta\mathbf{x})^\intercal \delta\mathbf{x}}{\delta\mathbf{x}^\intercal\,\delta\mathbf{x}}\delta\mathbf{x}, & \text{if } \delta\mathbf{x}\neq\mathbf{0}, \\ \nabla H(\mathbf{x}) & \text{otherwise}, \end{cases} \tag{35}$$

defines a discrete gradient which is naturally symmetric (see the lecture notes [17, (26-27)]).

Remark 8 (Equivalent discrete gradients)**.** *If a Hamiltonian H satisfies assumption $\mathbb{II}$, then all its discrete gradients coincide. This property is also satisfied for quadratic Hamiltonians, even with non-separated variables (see remark 9 below). In summary:*

$$\nabla_{\mathbb{III}}H = \nabla^{\pi}_{\mathbb{III}}H = \overline{\nabla_{\mathbb{III}}}H = \overline{\nabla_{mid}}H, \quad \textit{under assumption } (\mathbb{II}) \textit{ or if } H \textit{ is quadratic.}$$

As a corollary, if H is a sum of K functions, namely, $H(\mathbf{x}=[x_1,\dots,x_{N_1+\cdots+N_K}]^\intercal) = \sum_{k=1}^{K} H_k([x_{n_k+1},\dots,x_{n_k+N_k}]^\intercal)$ (with indexes $n_1=0$, $n_{k+1}=n_k+N_k$ and $N_k\geq 1$), then the average in (34) can be restricted to permutations operating separately inside the K subgroups of state variables. This advantageously involves $N_1!\dots N_K!$ rather than $(N_1+\cdots+N_K)!$ terms.

Remark 9 (Quadratic Hamiltonian and midpoint rule)**.** *For linear systems, the discrete gradient method corresponds to the midpoint rule. Indeed, for quadratic Hamiltonians, the continuous and discrete gradients are given by*

$$H(\mathbf{x}) = \tfrac{1}{2}\mathbf{x}^\intercal \boldsymbol{L}\mathbf{x} \textit{ with } \boldsymbol{L}=\boldsymbol{L}^\intercal \succ 0 \Longrightarrow \nabla H(\mathbf{x}) = \boldsymbol{L}\mathbf{x} \textit{ and } \overline{\nabla_D}H(\mathbf{x},\delta\mathbf{x}) = \boldsymbol{L}(\mathbf{x}+\tfrac{1}{2}\delta\mathbf{x}), \tag{36}$$

so that $\overline{\nabla_D}H(\mathbf{x},\delta\mathbf{x}) = \nabla H(\mathbf{x}_{mid})$ with $\mathbf{x}_{mid} = \mathbf{x}+\frac{1}{2}\delta\mathbf{x}$. This is not the case in general.

The definitions and properties of the various discrete gradients are summarised below.

Case on H	non symmetric ∇_D						symmetric $\overline{\nabla_D}$			
$\mathbb{II}$	$\nabla_{\mathbb{II}}$	$=$	$\nabla_{\mathbb{III}}$	$=$	$\nabla^{\pi}_{\mathbb{III}}$	$=$	$\overline{\nabla_{\mathbb{III}}}$	$=$	$\overline{\nabla_{\text{mid}}}$	
$\mathbb{III}$-quadratic	$\times$		$\nabla_{\mathbb{III}}$	$=$	$\nabla^{\pi}_{\mathbb{III}}$	$=$	$\overline{\nabla_{\mathbb{III}}}$	$=$	$\overline{\nabla_{\text{mid}}}$	($\equiv$ midpoint rule)
$\mathbb{III}$-general	$\times$		$\nabla_{\mathbb{III}}$	$\neq$	$\nabla^{\pi}_{\mathbb{III}}$	$\neq$	$\overline{\nabla_{\mathbb{III}}}$	$\neq$	$\overline{\nabla_{\text{mid}}}$	(all distinct)
Definition	(28)		(29)		(29,33)		(29,34)		(35)	

3.6 Method (implicit numerical scheme)

Consider a PHS described by the hybrid formulation (9) for sake of generality[15], where

$$\boldsymbol{M}[k] := \boldsymbol{M}_{\text{h}}\Big(\widehat{\mathbf{X}}[k],\widehat{\mathbf{W}}[k],\widehat{\mathbf{U}}[k]\Big), \quad \text{with} \quad \boldsymbol{M} := \boldsymbol{J}_{\text{h}} - \boldsymbol{R}_{\text{h}} \tag{37a}$$

is evaluated at samples k, choosing the basic (causal non time-symmetric) estimates

$$\widehat{\mathbf{X}}[k] := \mathbf{x}[k], \quad \widehat{\mathbf{W}}[k] := \mathbf{w}[k], \quad \widehat{\mathbf{U}}[k] := \mathbf{u}[k], \tag{37b}$$

[15] in the generic sense that $\mathbf{w}$ can be empty or $\boldsymbol{R}_{\text{h}}$ can be zero.

that do not involve any time-increments for simplicity (see remarks 10-11 for comments on properties). Then, for a discrete gradient ∇_D, the discrete-time system described by, for all $k \in \mathbb{Z}$ (or, given an initial condition $\mathbf{x}[0]$, for all $k \in \mathbb{N}$),

$$\begin{bmatrix} \delta\mathbf{x}[k]/\delta t \\ \mathbf{w}[k] \\ \mathbf{y}[k] \end{bmatrix} = \boldsymbol{M}[k] \begin{bmatrix} \nabla_D H\Big(\mathbf{x}[k],\ \delta\mathbf{x}[k]\Big) \\ \mathbf{z}\Big(\mathbf{w}[k]\Big) \\ \mathbf{u}[k] \end{bmatrix}, \tag{38a}$$

$$\mathbf{x}[k+1] = \mathbf{x}[k] + \delta\mathbf{x}[k], \tag{38b}$$

defines a discrete-time PHS that approximates (9) and fulfills the discrete power balance

$$\underbrace{(E[k+1]-E[k])/\delta t}_{P_{\mathbf{stored}}[k]} + \underbrace{\mathbf{z}(\mathbf{w}[k])^\intercal \mathbf{w}[k] + \mathbf{e}_\mathrm{h}[k]^\intercal \boldsymbol{R}_\mathrm{h}[k]\, \mathbf{e}_\mathrm{h}[k]}_{P_{\mathbf{diss}}[k] \geq 0} + \underbrace{\mathbf{u}[k]^\intercal \mathbf{y}[k]}_{P_{\mathbf{ext}}[k]} = 0. \tag{39}$$

Equations (38a-38b) define an implicit numerical scheme, since (38a) must be solved w.r.t. $(\delta\mathbf{x}[k], \mathbf{w}[k])$ at each sample k. The next section is devoted to avoid the use of (fixed-point, Newton-Raphson, etc.) iterative solvers.

Remark 10 (Accuracy)**.** *The accuracy order of this method is shown to be 1 in general, and 2 if* $\mathbf{w}$ *is empty and* $\boldsymbol{M}$ *is constant. Methods based on the discrete gradient, of order higher than* 1*, can be built using more refined estimates than (37b) that involve increments (implicit estimates, see e.g. [17, § 3]) or that involve multiples stages (with possibly non implicit estimates, see [26, 27, 49]).*

Remark 11 (About properties and features)**.** *In the context of PHS, important properties and features of numerical schemes are:* (a) preserving passivity and the power balance *(structured into conservative, dissipative/irreverible and external parts),* (b) the accuracy order *(error consistency, error stability, convergence), and possible additional features according to some objectives such as, for audio-acoustics,* (c) the computation cost *(with real time application perspectives),* (d) the rejection of aliasing *(artifact naturally produced when combining nonlinearities and sampling), or, some general properties such as (see the lecture notes [17, § 3])* (e) simplecticity *(area/hypervolume conservation in the phase space) of Hamiltonian flow*[16]*,* (f) preserving time-reversal symmetries *(see* $\overline{\boldsymbol{J}}$ *in [17, § 3.1]), etc. The method (38a-38b) achieves properties* (a) *(see (39)),* (b) *with a low but improvable order (remark 10) and prepares* (c) *(§ 4). Extensions to* (c,d) *with controllable orders are mentioned in the conclusion.*

3.7 Concluding remarks

Work on structure-preserving numerical methods is extensive [18]. Energy-preserving methods based on discrete gradient [19, 20] or average-vector field [21] in order to derive discrete-time PHS have been used in several works (see e.g. [22, 23] or e.g. [24, see also other papers of this author] for high order accuracy). Complementary references can be found in the lecture notes [17]. Recent work introducing RPM (for Regular Power-Balanced Methods) [30, chap. 5], which deals with the properties (a-d,f) with time-continuous frames, should also be mentioned.

[16] this property can be combined with (a), accepting to modify the original Hamiltonian.

4 Quadratisation method and non-iterative solver

Consider the problem (38a-38b) restricted to the differential formulation ($\mathbf{w}$ is empty) as (4) for simplicity of presentation: omitting $[k]$ for conciseness, (38a) is replaced by,

$$\begin{bmatrix} \delta\mathbf{x}/\delta t \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{M}_{\mathbf{xx}} & \boldsymbol{M}_{\mathbf{xu}} \\ \boldsymbol{M}_{\mathbf{yx}} & \boldsymbol{M}_{\mathbf{yu}} \end{bmatrix} \begin{bmatrix} \nabla_D H\Big(\mathbf{x},\, \delta\mathbf{x}\Big) \\ \mathbf{u} \end{bmatrix} \text{ with } \boldsymbol{M} = \boldsymbol{J} - \boldsymbol{R}. \tag{40}$$

4.1 Objective and principle

This section presents a numerical method with conditions under which the implicit numerical scheme (40) admits an explicit solution.

The objective is to avoid the use of iterative solvers such as the Newton-Raphson or the fixed-point algorithms. Indeed, for non-sparse low-dimensional problems, this alternative can be useful to reduce the computational cost and guarantee real-time for some audio-acoustics applications.

The principle introduced in [49] relies on (§ 4.2) the existence of such a solution for linear systems (quadratic hamiltonian) and (§ 4.3) the introduction of a change of variables that (§ 4.4) quadratises the hamiltonian while preserving the skew-symmetric (resp., non-negative) nature of matrix $\boldsymbol{J}$ (resp., $\boldsymbol{R}$), for a large class of nonlinear systems.

4.2 Case of a quadratic hamiltonian

Assume that

$$H(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\intercal \boldsymbol{L}\mathbf{x} \text{ with } \boldsymbol{L} = \boldsymbol{L}^\intercal \succ 0. \tag{41a}$$

Then (see remarks 8-9), all its discrete gradients coincide and $\nabla_D H(\mathbf{x}, \delta\mathbf{x}) = \boldsymbol{L}(\mathbf{x} + \frac{1}{2}\delta\mathbf{x})$. The implicit equation to solve w.r.t. $\delta\mathbf{x}$ (upper line in (40)) rewrites $\delta\mathbf{x}/\delta t = \boldsymbol{A}(\mathbf{x}+\frac{1}{2}\delta\mathbf{x}) + \boldsymbol{B}\mathbf{u}$ so that

$$\delta\mathbf{x}/\delta t = \boldsymbol{\Delta}^{-1}(\boldsymbol{A}\mathbf{x} + \boldsymbol{B}\mathbf{u}), \tag{41b}$$

$$\text{with } \boldsymbol{A} := \boldsymbol{M}_{\mathbf{xx}}\boldsymbol{L},\ \ \boldsymbol{B} := \boldsymbol{M}_{\mathbf{xu}},\ \text{ and where } \boldsymbol{\Delta} := \boldsymbol{I} - \tfrac{\delta t}{2}\boldsymbol{A} \tag{41c}$$

is invertible, since $\boldsymbol{\Delta}$ is the sum of a positive matrix ($\boldsymbol{I} + \frac{\delta t}{2}\boldsymbol{R}_{\mathbf{xx}} \succ 0$) and a skew-symmetric one ($-\frac{\delta t}{2}\boldsymbol{J}_{\mathbf{xx}}$). Equations (41b-41c) provide a non-iterative process to compute the increment $\delta\mathbf{x}$ at each sample k, from which are computed the output (lower line in (40)) and the state at sample $k+1$ (updating equation (38b)).

Remark 12 (Effort equation)**.** *For the effort ($\mathbf{e}_\mathbf{x} = \boldsymbol{L}\mathbf{x}$), the equation on increment $\delta\mathbf{e}_\mathbf{x}$ is the same as (41b) with matrices $\boldsymbol{A}_e := \boldsymbol{M}_{\mathbf{xx}}$, $\boldsymbol{B}_e := \boldsymbol{M}_{\mathbf{xu}}$ and $\boldsymbol{\Delta}_e := \boldsymbol{L}^{-1} - \frac{\delta t}{2}\boldsymbol{A}_e$. This equation may be preferred as it potentially requires fewer flops, $\boldsymbol{L}$ being constant.*

▷ Exercise 1 (Q3b – Q5).

4.3 Change of state (continuous time domain)

Consider a $\mathscr{C}^1$-regular bijective change of state[17] $\mathbf{Q}$ and the Hamiltonian H_q

$$\mathbf{Q}:\mathbf{x}\longmapsto\mathbf{q} \quad \text{of inverse denoted } \mathbf{X}:=\mathbf{Q}^{-1}:\mathbf{q}\longmapsto\mathbf{x}, \tag{42a}$$

$$H_q:\mathbf{q}\longmapsto H(\mathbf{x}=\mathbf{X}(\mathbf{q})), \tag{42b}$$

and denote the (invertible) associated Jacobian matrix (from $\mathbf{x}$ to $\mathbf{q}$)

$$\mathfrak{J}_{\mathbf{qx}}(\mathbf{x}):=\nabla_{\mathbf{x}}^{\mathsf{T}}\mathbf{Q}(\mathbf{x})=\begin{bmatrix}\partial_{x_1}Q_1(\mathbf{x}) & \dots & \partial_{x_N}Q_1(\mathbf{x})\\ \vdots & \vdots & \vdots\\ \partial_{x_1}Q_N(\mathbf{x}) & \dots & \partial_{x_N}Q_N(\mathbf{x})\end{bmatrix}. \tag{42c}$$

Then, the following differential formulations represent the same PHS:

$$\begin{bmatrix}\dot{\mathbf{x}}\\ \mathbf{y}\end{bmatrix}=\boldsymbol{M}(\mathbf{x},\mathbf{u})\begin{bmatrix}\nabla H(\mathbf{x})\\ \mathbf{u}\end{bmatrix}\Longleftrightarrow\begin{bmatrix}\dot{\mathbf{q}}\\ \mathbf{y}\end{bmatrix}=\boldsymbol{M}_q(\mathbf{q},\mathbf{u})\begin{bmatrix}\nabla H_q(\mathbf{q})\\ \mathbf{u}\end{bmatrix}\ \text{ with } \boldsymbol{M}_q:=\mathscr{Q}\boldsymbol{M}, \tag{43}$$

where the linear transformation $\mathscr{Q}$ defined by

$$[\mathscr{Q}\boldsymbol{M}](\mathbf{q},\mathbf{u}):=\mathfrak{J}_q(\mathbf{q})\,\boldsymbol{M}(\mathbf{X}(\mathbf{q}),\mathbf{u})\,\mathfrak{J}_q(\mathbf{q})^{\mathsf{T}}\text{ with }\mathfrak{J}_q(\mathbf{q}):=\begin{bmatrix}\mathfrak{J}_{\mathbf{qx}}\circ\mathbf{X}(\mathbf{q}) & \mathbf{0}\\ \mathbf{0} & \boldsymbol{I}_{\dim(\mathbf{u})}\end{bmatrix}, \tag{44}$$

preserves the skew-symmetry, the symmetry and the positivity of matrices, so that

$$\boldsymbol{M}=\boldsymbol{J}-\boldsymbol{R},\ \boldsymbol{J}=-\boldsymbol{J}^{\mathsf{T}},\ \boldsymbol{R}=\boldsymbol{R}^{\mathsf{T}}\succeq 0\ \xrightarrow{\mathscr{Q}}\ \boldsymbol{M}_q=\boldsymbol{J}_q-\boldsymbol{R}_q,\ \boldsymbol{J}_q=-\boldsymbol{J}_q^{\mathsf{T}},\ \boldsymbol{R}_q=\boldsymbol{R}_q^{\mathsf{T}}\succeq 0.$$

The proof stems from the mappings between the pairs of states, flows and efforts:

For all times, these pairs are related as

$$\text{(states)}\qquad \mathbf{q}=\mathbf{Q}(\mathbf{x})\ \ (\Leftrightarrow\mathbf{x}=\mathbf{X}(q)), \tag{45}$$

$$\text{(flows)}\qquad \dot{\mathbf{q}}=\mathfrak{J}_{\mathbf{qx}}(\mathbf{x})\,\dot{\mathbf{x}}, \tag{46}$$

$$\text{(efforts)}\qquad \nabla H(\mathbf{x})=\nabla_{\mathbf{x}}[H_q\circ\mathbf{Q}](\mathbf{x})=\mathfrak{J}_{\mathbf{qx}}(\mathbf{x})^{\mathsf{T}}\,[\nabla_{\mathbf{q}}H_q](\mathbf{Q}(\mathbf{x})). \tag{47}$$

Expressing the structure of $\boldsymbol{M}$ as in (4) and its dependency w.r.t. $(\mathbf{x},\mathbf{u})$, it comes

$$\begin{aligned}\dot{\mathbf{q}}&\overset{(46,4)}{=}\mathfrak{J}_{\mathbf{qx}}(\mathbf{x})\Big(\boldsymbol{M}_{\mathbf{xx}}(\mathbf{x},\mathbf{u})\nabla H(\mathbf{x})+\boldsymbol{M}_{\mathbf{xu}}(\mathbf{x},\mathbf{u})\mathbf{u}\Big)\\ &\overset{(47)}{=}\mathfrak{J}_{\mathbf{qx}}(\mathbf{x})\Big(\boldsymbol{M}_{\mathbf{xx}}(\mathbf{x},\mathbf{u})\mathfrak{J}_{\mathbf{qx}}(\mathbf{x})^{\mathsf{T}}\,[\nabla_{\mathbf{q}}H_q](\mathbf{Q}(\mathbf{x}))+\boldsymbol{M}_{\mathbf{xu}}(\mathbf{x},\mathbf{u})\mathbf{u}\Big)\\ &\overset{(45)}{=}\mathfrak{J}_{\mathbf{qx}}(\mathbf{X}(\mathbf{q}))\Big(\boldsymbol{M}_{\mathbf{xx}}(\mathbf{X}(\mathbf{q}),\mathbf{u})\mathfrak{J}_{\mathbf{qx}}(\mathbf{X}(\mathbf{q}))^{\mathsf{T}}\,[\nabla_{\mathbf{q}}H_q](\mathbf{q})+\boldsymbol{M}_{\mathbf{xu}}(\mathbf{X}(\mathbf{q}),\mathbf{u})\mathbf{u}\Big),\\ \mathbf{y}&\overset{(47,45)}{=}\boldsymbol{M}_{\mathbf{yx}}(\mathbf{X}(\mathbf{q}),\mathbf{u})\mathfrak{J}_{\mathbf{qx}}(\mathbf{X}(\mathbf{q}))^{\mathsf{T}}\,[\nabla_{\mathbf{q}}H_q](\mathbf{q})+\boldsymbol{M}_{\mathbf{yu}}(\mathbf{X}(\mathbf{q}),\mathbf{u})\mathbf{u},\end{aligned}$$

yielding (43-44). The preservation of the symmetries and of the spectrum sign of a matrix $\boldsymbol{K}$ through $\mathscr{Q}$ stems from its form in $\mathfrak{J}_q\,\boldsymbol{K}\,\mathfrak{J}_q^{\mathsf{T}}$ with $\mathfrak{J}_q$ invertible.

[17] We consider here diffeomorphisms $\mathbf{Q}:\mathbb{R}^N\to\mathbb{R}^N$.

4.4 Quadratisation method and solver

A port-Hamiltonian system (4) with a $\mathscr{C}^1$-regular hamiltonian $H(\mathbf{x})$ is said to be quadratisable if it exists a $\mathscr{C}^1$-regular bijective change of state $\mathbf{Q} = \mathbf{X}^{-1}$ such that

$$H_q(\mathbf{q}) := H \circ \mathbf{X}(\mathbf{q}) = \frac{1}{2}\mathbf{q}^T \boldsymbol{L}\mathbf{q}, \text{ with } \boldsymbol{L} \succ 0. \tag{48}$$

Then, the $\mathbf{q}$-representation of the PHS (right-hand side of equivalence (43)) has a quadratic hamiltonian so that the non-iterative solver proposed in section 4.2 applies to this $\mathbf{q}$-representation

$$\begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{y} \end{bmatrix} = \boldsymbol{M}_q(\mathbf{q},\mathbf{u}) \begin{bmatrix} \boldsymbol{L}\mathbf{q} \\ \mathbf{u} \end{bmatrix} \text{ with } \boldsymbol{M}_q := \boldsymbol{J}_q - \boldsymbol{R}_q,\ \boldsymbol{J}_q^\intercal = \boldsymbol{J}_q,\ \boldsymbol{R}_q = \boldsymbol{R}_q^\intercal \succeq 0. \tag{49}$$

An important corollary is the existence and uniqueness of the discrete-time solution. The quadratisation method can be summarised in the following sequence of steps.

Procedure:

Given a PHS under formulation (4) with hamiltonian $H(\mathbf{x})$:

1. Find $\mathbf{Q}$ such that (48) is satisfied,
2. Build (=derive or implement) function $\mathbf{X} := \mathbf{Q}^{-1}$,
3. Build matrices $\boldsymbol{M}_q := \boldsymbol{J}_q - \boldsymbol{R}_q,\ \boldsymbol{J}_q^\intercal = \boldsymbol{J}_q,\ \boldsymbol{R}_q = \boldsymbol{R}_q^\intercal$ as functions of $(\mathbf{q},\mathbf{u})$, by applying transformation $\mathscr{Q}$ (see (44)),
4. Build matrices $\boldsymbol{A}_q$, $\boldsymbol{B}_q$ and $\boldsymbol{\Delta}_q$ as functions of $(\mathbf{q},\mathbf{u})$ for the $\mathbf{q}$-representation (49) following § 4.2 (adapting labels q instead of x in (41c)),
5. Build the incremental state function $\mathbf{q}[k]$ from (41b),
6. Build the updating equation $\mathbf{q}[k+1] = \mathbf{q}[k] + \delta\mathbf{q}[k]$ and the observation equation on $\mathbf{y}$.

→ Your input-output power-balanced simulation is (almost) ready!
→ The state $\mathbf{x}$ can also be evaluated with $\mathbf{x} = \mathbf{X}(\mathbf{q})$.

Remark 13 (Units)**.** *In this definition, matrix $\boldsymbol{L}$ can be chosen such that $\mathbf{q}$ has the same physical units as $\mathbf{x}$ (or other properly-chosen physical units). It can also be chosen, without loss of generality, as the dimensionless identity matrix, meaning that the units of $\mathbf{q}$ are square-roots of Joules [$\sqrt{\mathrm{J}} = \sqrt{\mathrm{Kg}}.\mathrm{m}.\mathrm{s}^{-1}$].*

Step 1 is straightforward for the class of Hamiltonians such that ($1 \leq n \leq N$, $N \geq 1$)

$$\text{(separated variables, H1)}: \quad H(\mathbf{x} := [x_1, \ldots, x_N]^\intercal) = \sum_{n=1}^{N} H_n(x_n), \tag{50a}$$

$$(\mathscr{C}^1\text{-regularity, H2}): \quad H_n \text{ is } \mathscr{C}^1\text{-regular}, \tag{50b}$$

$$\text{(locally quadratic at 0, H3)}: \quad H_n(x) \underset{0}{\sim} \frac{k_n}{2} x^2 \text{ with } k_n > 0, \tag{50c}$$

$$\text{(strict quasi-convexity, H4)}: \quad \forall x \neq y,\ H_n(\lambda x + (1-\lambda) y) < \max(H_n(x), H_n(y)). \tag{50d}$$

Under hypotheses (H1-4), function

$$\mathbf{Q}(\mathbf{x}) := [q_1(x_1), \ldots q_N(x_N)]^\intercal \text{ with } q_n(x_n) := \operatorname{sign}(x_n) \sqrt{2H_n(x_n)}, \tag{51}$$

defines a $\mathscr{C}^1$-regular bijection that satisfies (48) with $\boldsymbol{L} = \boldsymbol{I}$.

Proof:

• Function H_n is continuous (H2), null at 0 (H3) and then, from (H4), non-negative strictly decreasing (resp. increasing) on $\mathbb{R}^-$ (resp. $\mathbb{R}^+$). It follows from (51) that q_n is a real-valued strictly increasing continuous function.

• The $\mathscr{C}^1$-regularity of q_n is obvious on $\mathbb{R} \setminus \{0\}$ (H2 and $H(x) > 0$), where $q_n'(x) = f_n(x) := \sqrt{2} H_n'(x) / (2\sqrt{H_n(x)}) > 0$ if $x \in (0, +\infty)$ and $q_n'(x) = -f_n(x) > 0$ if $x \in (-\infty, 0)$. It is also satisfied at 0 since, from (H3), $f_n(x) \underset{0}{\sim} \sqrt{2} k_n x / (2\sqrt{k_n x^2/2}) = \sqrt{k_n} \operatorname{sign}(x)$ so that $\lim_{x \to 0^+} q_n'(x) = +\sqrt{k_n} = \lim_{x \to 0^-} q_n'(x)$.

• Finally, $\mathbf{Q}$ is a collection of $\mathscr{C}^1$-regular bijections, that concludes the proof.

Remark 14 (non-separated variables)**.** *In the case of Hamiltonians with non-separated variables, sufficient conditions for the quadratisation can be found in [26, Property 16, Example 19] and [27, Property 2.9].*

Remark 15 (higher accuracy order)**.** *Following remark 10, due to the* $\mathbf{q}$*-dependency of* $\boldsymbol{J}_q$ *and* $\boldsymbol{R}_q$ *the discrete gradient method is accurate only at order 1 (also with the quadratisation method). Higher-order methods that benefit from the quadratisation are available in [49, 26, 27]: they involve multiple stages to refine the estimates and increments in (37a-38b), inspired from Runge-Kutta methods. These methods achieve properties (a-c) of remark 11 for orders 1, 2 or 3 without iterative solver (see [26, p.115-122]).*

Note that after this method was introduced in [49], an alternative method (also based on quadratisation but with auxiliary variables) was proposed in [25]. These methods are used in [6, 7] for in audio-acoustic applications.

5 Exercises

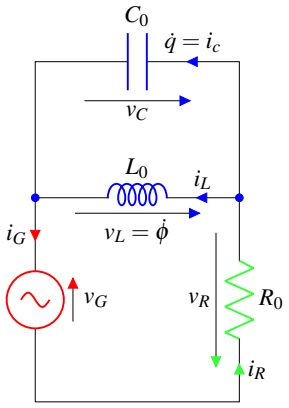

Figure 1: Circuit and notations for exercises

5.1 Exercise 1 (linear circuit): formulations and power-balanced simulation

Consider the circuit, with notations and conventions[18] presented in figure 1: $q(t)$ denotes the charge of a capacitor with linear effort law $q = C v_C$; $\phi(t)$ denotes the magnetic flux of a coil with linear effort law $\phi = L i_L$.

Question 1. *Differential Algebraic formulation.*
*(a) Define the components (***x***,H), (w,z), (y,u).* Suggestion: *choose* $\mathbf{x} = [q, \phi]^\intercal$, $w = v_R$ *and* $u = v_G$.
(b) Detail how the flows $\mathbf{f} := [\dot{\mathbf{x}}^T, w, y]^\intercal$ *and efforts* $\mathbf{e} = [\nabla H(\mathbf{x})^\intercal, z(w), u]^\intercal$ *identify with the physical quantities (currents or voltages) of figure 1.*
(c) Derive the interconnection matrix $\boldsymbol{S}$ *from the Kirchhoff laws and write the PHS in the form (1).*

Question 2. *Differential formulation*
(a) Express $\boldsymbol{\Gamma}$ *and matrix* $\boldsymbol{P}$ *as in § 2.3. Are the conditions (6a-6b) satisfied ?*
(b) Express $\boldsymbol{J}_\Gamma$ *and* $\boldsymbol{J}_\mathrm{D}$, $\boldsymbol{R}_\Gamma$ *and* $\boldsymbol{R}_\mathrm{D}$, *from (8a-8b).*
(c) Write the differential formulation (4).

Observe (a posteriori) that you could find this result directly from the circuit: the interest of the procedure (a,b,c) is its systematic property.

Question 3. *Discrete gradient and numerical method.*
(a) Derive the discrete gradient associated with the Hamiltonian H using (28).
(b) Observe that it also satisfies the property (36) in remark 9 (valid for all quadratic Hamiltonians), so that the method in § 4.2 applies.

[18] Note that the receiver convention is used for all dipoles (currents and voltages are oppositely oriented, including for the voltage generator).

Question 4. *Simulation.*
Implement a code based on § 4.2, considering initial conditions and an input signal u, with parameters δt and R_0, L_0 C_0 [SI units].

Suggestion: *to reduce flops, express matrix $\boldsymbol{R}$ as the product of a scalar parameter and a coefficient-free matrix.*

Question 5. *Numerical experiments.*
Remark: *a LC circuit is a (conservative) oscillator with natural period $T_0 = 2\pi\sqrt{L_0 C_0}$.*
(a) Visualise simulations in the convenient situation, where oscillations are slower than the Shannon-Nyquist period ($T_0 > 2\delta t$).
(b) Test simulations in free regime ($u = 0$) for a nonzero initial condition for various values of $R_0 > 0$. Observe the energy signal and the numerical error on the power balance $\eta := P_{\mathbf{stored}} + P_{\mathbf{diss}} + P_{\mathbf{ext}}$.
(c) Test the robustness of the numerical passivity to any $\delta t > 0$.

Suggestion: *Plots to consider are (1) signals involved in* $\mathbf{x}$, $\mathbf{f}$, $\mathbf{e}$*; (2) energy $H(\mathbf{x})$; (3) power signals $P_{\mathbf{stored}}$, $P_{\mathbf{diss}}$, $P_{\mathbf{ext}}$ (on a same graph); (4) the numerical error η on the power balance.*

Plots to consider are (1) signals involved in $\mathbf{x}$, $\mathbf{f}$, $\mathbf{e}$*; (2) energy $H(\mathbf{x})$; (3) power signals $P_{\mathbf{stored}}$, $P_{\mathbf{diss}}$, $P_{\mathbf{ext}}$ (on a same graph); (4) the numerical error η on the power balance.*

Remark: *observe that the conservative case ($R_0 = 0$) makes the representations derived in questions 1-2 degenerate in a singular way. Addressing this case requires to include constraints in the algebraic-differential formulations or to consider kernel representations of Dirac structures (see footnote 10 in § 2.6).*

5.2 Exercise 2 (nonlinear circuit): quadratisation method

Consider the same circuit with a coil characterised by a saturating effort law of Hamiltonian $H(\phi) = E_0 \ln(\cosh(\phi/\phi_0))$.

Question 6. *PHS.*
(a) How to modify the formulations (1) and (4) ? (b) What is the effort law of the coil ?
(c) What is the equivalent inductance L_0 for small amplitudes ?

Question 7. *Quadratised PHS.*
Derive the quadratised PHS by applying the procedure of § 4.4 (steps 1-4).

Question 8. *Simulation and numerical experiments.*
Implement a code (by adapting that of exercise 1).

Suggestion: *to appreciate the nonlinear effect and simplify comparisons with exercise 1, choose ϕ_0 smaller than the maximal amplitude of your previous simulations, then fix E_0 to have the same L_0 as your previous code.*

5.3 Exercise 3 (nonlinear circuit): PHS shifting

(This exercise can be done after question 6)

Consider that the generator has a bias: $v_G(t) = v_G^\star + \widetilde{v_G}(t)$

Question 9. *Write the equations satisfied by* $\mathbf{x}^\star$. *Is the solution unique ?*

Question 10. *Derive the shifted Hamiltonian* $\widetilde{H_{\mathbf{x}^\star}}(\widetilde{\mathbf{x}})$ *(see (19)).*

Question 11. *Can the shifted-PHS be quadratised ? Propose an implementation.*

5.4 Exercise 4 (pyPHS)

The Python library pyPHS provides tools to automate the derivation of the PHS (in LaTeX), of the discrete gradient-based numerical scheme with a Newton-Raphson solver, and to generate C++ code for simulation. After this lecture, the reader is invited to follow the presentation, tutorials and examples (see link indications in [77]).

References

[1] Bernhard M Maschke and Arjan J van der Schaft. Port-controlled hamiltonian systems: modelling origins and systemtheoretic properties. In *Nonlinear Control Systems Design 1992*, pages 359–365. Elsevier, 1993.

[2] Arjan Van Der Schaft. Port-hamiltonian systems: an introductory survey. In *Proceedings of the international congress of mathematicians*, volume 3, pages 1339–1365. Citeseer, 2006.

[3] Vincent Duindam, Alessandro Macchelli, Stefano Stramigioli, and Herman Bruyninckx. *Modeling and control of complex physical systems: the port-Hamiltonian approach*. Springer Science & Business Media, 2009.

[4] Arjan Van Der Schaft and Dimitri Jeltsema. Port-hamiltonian systems theory: An introductory overview. *Foundations and Trends in Systems and Control*, 1(2-3):173–378, 2014.

[5] Antoine Falaize. Simulation of finite-dimensional multi-physical systems described by networks of components. In *Newsletter of the European Community on Computational Methods in Applied Sciences, ECCOMAS*, pages 49–53, 2020.

[6] Mohammed Danish, Stefan Bilbao, and Michele Ducceschi. Applications of port hamiltonian methods to non-iterative stable simulations of the korg35 and moog 4-pole vcf. In *Proceedings of the International Conference on Digital Audio Effects*, 2021.

[7] Michele Ducceschi and Stefan Bilbao. Simulation of the geometrically exact nonlinear string via energy quadratisation. *arXiv preprint arXiv:2112.15008*, 2021.

[8] Luis Mora Araque. *Port-Hamiltonian modeling of fluid-structure interactions in a longitudinal domain*. PhD thesis, Université Bourgogne Franche-Comté; Universidad técnica Federico Santa María . . . , 2020.

[9] Martin Pfeifer, Sven Caspart, Silja Pfeiffer, Charles Muller, Stefan Krebs, and Soeren Hohmann. Automated generation of explicit port-hamiltonian models from multi-bond graphs. *arXiv preprint arXiv:1909.02848*, 2019.

[10] Martin Pfeifer. *Automated Model Generation and Observer Design for Interconnected Systems : a Port-Hamiltonian Approach*. PhD thesis, Karlsruher Institut für Technologie (KIT), 2021.

[11] Markus Lohmayer and Sigrid Leyendecker. Ephs: A port-hamiltonian modelling language. *arXiv preprint arXiv:2202.00377*, 2022.

[12] Arjan van der Schaft and Bernhard Maschke. Generalized port-hamiltonian dae systems. *Systems & Control Letters*, 121:31–37, 2018.

[13] Arjan van der Schaft and Bernhard Maschke. Dirac and lagrange algebraic constraints in nonlinear port-hamiltonian systems. *Vietnam Journal of Mathematics*, 48(4):929–939, Dec 2020.

[14] Arjan Van der Schaft. *L2-gain and passivity techniques in nonlinear control*. Springer, 2000.

[15] Bayu Jayawardhana, Romeo Ortega, Eloisa Garcia-Canseco, and Fernando Castanos. Passivity of nonlinear incremental systems: Application to pi stabilization of nonlinear rlc circuits. *Systems & control letters*, 56(9-10):618–622, 2007.

[16] Claudio De Persis and Nima Monshizadeh. Bregman storage functions for microgrid control. *IEEE Transactions on Automatic Control*, 63(1):53–68, 2017.

[17] Paul Kotyczka and Laurent Lefèvre. Discrete-time port-hamiltonian systems and control. In *2nd Spring School on Theory and Applications of Port-Hamiltonian Systems, Fauenchiemsee*, pages 1–15, 2022.

[18] Ernst Haier, Christian Lubich, and Gerhard Wanner. *Geometric Numerical integration: structure-preserving algorithms for ordinary differential equations*. Springer, 2006.

[19] Toshiaki Itoh and Kanji Abe. Hamiltonian-conserving discrete canonical equations based on variational difference quotients. *Journal of Computational Physics*, 76(1):85–102, 1988.

[20] O. Gonzalez. Time integration and discrete hamiltonian systems. *Journal of Nonlinear Science*, 6(5):449, 1996.

[21] GRW Quispel and David Ian McLaren. A new class of energy-preserving numerical integration methods. *Journal of Physics A: Mathematical and Theoretical*, 41(4):045206, 2008.

[22] Saïd Aoues. *Schémas d'intégration dédiés à l'étude, l'analyse et la synthèse dans le formalisme Hamiltonien à ports*. PhD thesis, INSA de Lyon, 2014.

[23] Said Aoues, Michael Di Loreto, Damien Eberard, and Wilfrid Marquis-Favre. Hamiltonian systems discrete-time approximation: Losslessness, passivity and composability. *Systems & Control Letters*, 110:9–14, 2017.

[24] Elena Celledoni and Eirik Hoel Høiseth. Energy-preserving and passivity-consistent numerical discretization of port-hamiltonian systems. *arXiv preprint arXiv:1706.08621*, 2017.

[25] Xiaofeng Yang, Jia Zhao, and Qi Wang. Numerical approximations for the molecular beam epitaxial growth model based on the invariant energy quadratization method. *Journal of Computational Physics*, 333:104–127, 2017.

B. References related to the S3AM team (STMS-lab, IRCAM-CNRS-SU)

PHD theses (manuscripts and videos of the public defenses)

[26] Nicolas Lopes. *Approche passive pour la modélisation, la simulation et l'étude d'un banc de test robotisé pour les instruments de type cuivre*. PhD thesis, Université Paris 6 (UPMC), Jun 2016. thesis defense video: https://medias.ircam.fr/x2c52cb.

[27] Antoine Falaize. *Modélisation, simulation, génération de code et correction de systèmes multi-physiques audios : approche par réseau de composants et formulation Hamiltonienne à Ports*. Thesis, Université Pierre & Marie Curie - Paris 6, July 2016. thesis defense video: https://medias.ircam.fr/x6b7424.

[28] Tristan Lebrun. *Modélisation multi-physique passive, identification, simulation, correction et asservissement de haut-parleur sur des comportements cibles*. Thesis, Sorbonne Université, December 2019. confidential.

[29] Marc Wijnand. *Contrôle en temps fini de systèmes vibratoires hybrides couplant équations aux dérivées partielles et équations aux dérivées ordinaires : les cas du tom et du câble pesant*. Thesis, Sorbonne Université, July 2021. thesis defense video: https://www.youtube.com/watch?v=bTHff07CE_8.

[30] Rémy Müller. *Time-continuous power-balanced simulation of nonlinear audio circuits: realtime processing framework and aliasing rejection*. Thesis, Sorbonne Université, July 2021. thesis defense video: https://medias.ircam.fr/x773840.

[31] Victor Wetzel. *Lumped Power-Balanced Modelling and Simulation of the Vocal Apparatus: A Fluid-Structure-Interaction Approach*. Thesis, Sorbonne Université, December 2021. thesis defense video: https://www.youtube.com/watch?v=f0rJ8N-eE_0.

Journal papers

[32] Denis Matignon and Thomas Hélie. A class of damping models preserving eigenspaces for linear conservative port-Hamiltonian systems. *European Journal of Control*, 19(6):486–494, 2013.

[33] Antoine Falaize and Thomas Hélie. Passive Guaranteed Simulation of Analog Audio Circuits: A Port-Hamiltonian Approach. *Applied Sciences*, 6:273 – 273, 2016.

[34] Nicolas Lopes and Thomas Hélie. Energy Balanced Model of a Jet Interacting With a Brass Player's Lip. *Acta Acustica united with Acustica*, 102(1):141–154, 2016.

[35] Antoine Falaize and Thomas Hélie. Passive simulation of the nonlinear port-Hamiltonian modeling of a Rhodes Piano. *Journal of Sound and Vibration*, 390:289–309, March 2017.

[36] Judy Najnudel, Thomas Hélie, and David Roze. Simulation of the Ondes Martenot Ribbon-Controlled Oscillator Using Energy-Balanced Modeling of Nonlinear Time-Varying Electronic Components. *AES - Journal of the Audio Engineering Society Audio-Accoustics-Application*, 67(12):961–971, December 2019.

[37] Judy Najnudel, Thomas Hélie, David Roze, and Henri Boutin. Simulation of an ondes Martenot circuit. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 28:2651–2660, September 2020.

[38] Judy Najnudel, Thomas Hélie, David Roze, and Müller Rémy. Power-Balanced Modeling of Nonlinear Coils and Transformers for Audio Circuits. *AES - Journal of the Audio Engineering Society Audio-Acoustics-Application*, 69(7/8):506–516, July 2021.

[39] Antoine Falaize and Thomas Hélie. Passive modelling of the electrodynamic loudspeaker: from the Thiele–Small model to nonlinear port-Hamiltonian systems. *Acta Acustica*, 4(1):1, February 2020.

Conference papers

[40] Denis Matignon and Thomas Hélie. On damping models preserving the eigenfunctions of conservative systems: a port-Hamiltonian perspective. In *4th IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*, pages 1–6, Bertinoro, Italy, August 2012.

[41] Antoine Falaize and Thomas Hélie. Simulation of an analog circuit of a wah pedal: a port-Hamiltonian approach. In *135th convention of the Audio Engineering Society*, pages –, New-York, United States, October 2013.

[42] Antoine Falaize and Thomas Hélie. Modélisation d'un haut parleur électrodynamique: approche dans le cadre des Systemes à Hamitoniens à Ports. In *12e Congrès Français d'Acoustique*, Poitiers, France, April 2014.

[43] Nicolas Lopes and Thomas Hélie. Modèle d'interaction Jet/Lèvre préservant le bilan de puissance pour les instruments de type cuivre. In *12e Congrès Français d'Acoustique*, Poitiers, France, April 2014.

[44] Antoine Falaize and Thomas Hélie. Passive simulation of electrodynamic loudspeakers for guitar amplifiers: a port- Hamiltonian approach. In *International Symposium on Musical Acoustics*, pages 1–5, Le Mans, France, July 2014.

[45] Antoine Falaize, Nicolas Lopes, Thomas Hélie, Denis Matignon, and Bernhard Maschke. Energy-balanced models for acoustic and audio systems: a port-Hamiltonian approach. In *Unfold Mechanics for Sounds and Music*, pages 1–1, Paris, France, September 2014.

[46] Antoine Falaize and Thomas Hélie. Guaranteed-passive simulation of an electromechanical piano: A port-Hamiltonian approach. In *18th Int. Conference on Digital Audio Effects (DAFx-15)*, Trondheim, Norway, November 2015.

[47] Thomas Hélie, Antoine Falaize, and Nicolas Lopes. Systèmes Hamiltoniens à Ports avec approche par composants pour la simulation à passivité garantie de problèmes conservatifs et dissipatifs. In *Colloque National en Calcul des Structures*, volume 12, Giens, France, May 2015.

[48] Thomas Hélie and Denis Matignon. Nonlinear damping models for linear conservative mechanical systems with preserved eigenspaces: a port-Hamiltonian formulation. In *Lagrangian and Hamiltonian Methods for Non Linear Control*, Lyon, France, July 2015. IFAC.

[49] Nicolas Lopes, Thomas Hélie, and Antoine Falaize. Explicit second-order accurate method for the passive guaranteed simulation of port-Hamiltonian systems. In *5th IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control LHMNC 2015*, volume 48 of *IFAC-PapersOnLine*, pages 223–228, Lyon, France, July 2015. IFAC.

[50] Antoine Falaize, Nicolas Papazoglou, Thomas Hélie, and Nicolas Lopes. Compensation of loudspeaker's nonlinearities based on flatness and port-Hamiltonian approach. In *22ème Congrès Français de Mécanique*, Lyon, France, August 2015. Association Française de Mécanique.

[51] Thomas Hélie and Denis Matignon. Physically-based dynamic morphing of beam sounds. In *Vienna Talk on Music Acoustics*, volume 13, Vienna, Austria, September 2015.

[52] Thomas Hélie and David Roze. Corde non linéaire amortie : formulation hamiltonienne à ports, réduction d'ordre exacte et simulation à passivité garantie. In *13ème Congrès Français d'Acoustique*, Le Mans, France, April 2016.

[53] Nicolas Lopes and Thomas Hélie. Modélisation et simulation à passivité garantie d'un cuivre. In *Congrès Français d'Acoustique*, Le Mans, France, April 2016.

[54] Marguerite Jossic, David Roze, Thomas Hélie, Baptiste Chomette, and Adrien Mamou-Mani. Energy shaping of a softening Duffing oscillator using the formalism of Port-Hamiltonian Systems. In *20th International Conference on Digital Audio Effects (DAFx-17)*, Edinburgh, United Kingdom, September 2017.

[55] Rémy Muller and Thomas Hélie. Trajectory Anti-Aliasing on Guaranteed-Passive Simulation of Nonlinear Physical Systems. In *20th International Conference on Digital Audio Effects (DAFx-17)*, Edinburgh, United Kingdom, September 2017.

[56] Thomas Hélie and Fabrice Silva. Self-oscillations of a vocal apparatus: a port-Hamiltonian formulation. In Frank Nielsen and Frédéric Barbaresco, editors, *Geometric Science of Information: Third International Conference, GSI 2017, Paris, France, November 7-9, 2017, Proceedings*, 3rd conference on Geometric Science of Information (GSI), pages 375–383. Springer International Publishing, 2017.

[57] Antoine Falaize and Thomas Hélie. PyPHS: Un module Python pour la modélisation et la simulation à passivité garantie de systèmes multi-physiques. In *Rencontres Maths-Industrie en Acoustique Numérique et Signal Audio, Ecole Polytechnique*, Palaiseau, France, March 2017. Matthieu Aussal, Richard Fontanges, François Alouges.

[58] David Roze and Thomas Hélie. Simulation passive d'un modèle réduit exact de plaque de Berger en grandes déformations. In *14ème Congrès Français d'Acoustique*, Le Havre, France, April 2018.

[59] Fabrice Silva and Thomas Hélie. CFA2018/165 Modélisation physique, simulation à bilan de puissance garanti et examen de régimes d'un appareil vocal simplifié. In *14ème Congrès Français d'Acoustique*, Le Havre, France, April 2018.

[60] Marc Wijnand, Brigitte d'Andréa-Novel, Thomas Hélie, and David Roze. Contrôle des vibrations d'un oscillateur passif : stabilisation en temps fini et par remodelage d'énergie. In *14ème Congrès Français d'Acoustique*, Le Havre, France, April 2018.

[61] Rémy Muller and Thomas Hélie. Power-Balanced Modelling Of Circuits As Skew Gradient Systems. In *21st International Conference on Digital Audio Effects (DAFx-18)*, Aveiro, Portugal, September 2018.

[62] Judy Najnudel, Thomas Hélie, Henri Boutin, David Roze, Thierry Maniguet, and Stéphane Vaiedelich. Analog circuits and Port-Hamiltonian realizability issues: a resolution method for simulations via equivalent components. In *145th AES Convention*, New-York, United States, October 2018.

[63] Thomas Hélie, Fabrice Silva, and Victor Wetzel. Port-Hamiltonian approach to self-sustained oscillations in the vocal apparatus. In *NODYCON 2019 (Nonlinear Dynamics Conference)*, Rome, Italy, February 2019.

[64] Tristan Lebrun, Marc Wijnand, Thomas Hélie, David Roze, and Brigitte d'Andréa-Novel. Electroacoustic absorbers based on passive finite-time control of loudspeakers: a numerical investigation. In *International Nonlinear Dynamics Conference*, Rome, Italy, February 2019.

[65] Victor Wetzel, Thomas Hélie, and Fabrice Silva. Power balanced time-varying lumped parameter model of a vocal tract: modelling and simulation. In *26th International Conference on Sound and Vibration*, Montréal, Canada, July 2019. IIAV.

[66] Fabrice Silva, Thomas Hélie, and Wetzel Victor. Port-Hamiltonian Representation of Dynamical Systems. Application to Self-Sustained Oscillations in the Vocal Apparatus. In Publications du LMA, editor, *7th Int. Conf. on Nonlinear Vibrations, Localization and Energy Transfer*, volume 160 of *7 th International Conference on Nonlinear Vibrations, Localization and Energy Transfer*, ISBN 978–2–909669–26–7, Marseille, France, June 2019.

[67] Rémy Müller and Thomas Hélie. A minimal passive model of the operational amplifier: application to Sallen–Key analog filters. In *22nd International Conference on Digital Audio Effects (DAFx-19)*, Birmingham, France, August 2019.

[68] Marc Wijnand, Brigitte d'Andréa-Novel, Benoît Fabre, Thomas Hélie, Lionel Rosier, and David Roze. Active control of the axisymmetric vibration modes of a tom-tom drum. In *58th IEEE Conference on Decision and Control*, 2019 IEEE 58th Conference on Decision and Control (CDC), pages 6887–6892, Nice, France, December 2019. IEEE.

[69] Victor Wetzel, Thomas Hélie, and Fabrice Silva. Power-balanced modelling of the vocal tract: a recast of the classical lumped-parameter model. In *Forum Acusticum 2020 (e-Forum Acusticum)*, pages 1–8, Lyon, France, December 2020.

[70] Marc Wijnand, Brigitte D'Andréa-Novel, Thomas Hélie, and David Roze. Active control of the axisymmetric vibration modes of a tom-tom drum using a modal-based observer-regulator. In *Forum Acusticum*, pages 639–646, Lyon, France, December 2020.

[71] Rémy Müller and Thomas Hélie. Fully-implicit algebro-differential parametrization of circuits. In *23rd International Conference on Digital Audio Effects (DAFx-20)*, Vienne, Austria, September 2020.

[72] Judy Najnudel, Rémy Muller, Thomas Hélie, and David Roze. A power-balanced dynamic model of ferromagnetic coils. In *23rd International Conference on Digital Audio Effects (eDAFx-20)*, Proceedings of the 23rd International Conference on Digital Audio Effects (eDAFx2020), Vienne, Austria, September 2020.

[73] Fabrice Silva, Thomas Hélie, and Victor Wetzel. Energy-consistent modelling of the fluid-structure interaction in the glottis. In *12th International Conference on Voice Physiology and Biomechanics (ICVPB2020)*, ICVPB2020 proceedings Edition December 2020, Grenoble (on line), France, December 2020.

[74] Antoine Falaize and David Roze. A generic passive-guaranteed structure for elastoplatic friction models. In *Second International Nonlinear Dynamics Conference (NODYCON21)*, Italy, February 2021.

[75] Judy Najnudel, Thomas Hélie, David Roze, and Müller Remy. From statistical physics to macroscopic port-Hamiltonian Systems: A roadmap. In *7th IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*, Berlin, Germany, October 2021.

[76] Judy Najnudel, Müller Remy, Thomas Hélie, and David Roze. Identification of nonlinear circuits as port-Hamiltonian systems. In *24th International Conference on Digital Audio Effects (DAFx-21)*, Proceedings of the 24th International Conference on Digital Audio Effects (DAFx20in21), Vienna, Austria, September 2021.

Software

[77] Antoine Falaize and Thomas Hélie. PyPHS: Passive modeling and simulation in python, 2016.
Python package on github:
- https://github.com/pyphs/pyphs
Complementary web pages:
- https://pyphs.github.io/pyphs/ (more information on the algebraic-differential structure and the numerical method)
- https://github.com/pyphs/pyphs/tree/master/pyphs/examples (many examples)
- https://medias.ircam.fr/x152ab3 (presentation video in French):
- https://medias.ircam.fr/x4d31af (tutorial video in French).

Patent

[78] Thomas Hélie and Tristan Lebrun. Dispositif de génération d'un signal de commande d'un système électrique. Patent (Déclaration d'invention - INPI - FR1913122),
https://patentscope.wipo.int/search/fr/detail.jsf?docId=WO2021099509.

Acknowledgements

The author thanks (by alphabetical order) Ivan Cohen, Jean-Baptiste Dakeyo, Brigitte d'Andréa-Novel, Antoine Deschamps, Antoine Falaize, Thibault Geoffroy, Thomas Guennoc, Marguerite Jossic, Tristan Lebrun, Nicolas Lopes, Bernhard Maschke, Denis Matignon, Rémy Müller, Judy Najnudel, Nicolas Papazoglou, Mathis Raibaud, David Roze, Fabrice Silva, Tarik Usciati, Colette Voisembert, Victor Wetzel and Marc Wijnand for their collaborative work and publications on Port-Hamiltonian Systems within the S3AM team at STMS laboratory (IRCAM-CNRS-SU).