Foundations and Trends® in Robotics
Vol. 6, No. 3 (2017) 140–210

DOI: 10.1561/2300000038

now
the essence of knowledge

Energy in Robotics

Gerrit A. Folkertsma
Robotics and Mechatronics group
CTIT Institute University of Twente
g.a.folkertsma@ieee.org

Stefano Stramigioli
Robotics and Mechatronics group
CTIT/MIRA Institutes University of Twente
s.stramigioli@ieee.org

Contents

Abstract

Energy and energy exchange govern interactions in the physical world. By explicitly considering the energy and power in a robotic system, many control and design problems become easier or more insightful than in a purely signal-based view. We show the application of these energy considerations to robotics; starting from the fundamental aspects, but, most importantly, continuing to the practical application to robotic systems. Using the theory of Port-Hamiltonian Systems as a fundamental basis, we show examples concerning energy measurement, passivity and safety. Control by interconnection covers the shaping and directing of energy inside the controller algorithms, to achieve desired behaviour in a power-consistent manner. This idea of control over the energy flows is extended to the physical domain. In their mathematical description and analysis, the boundary between controller and robot disappears and everything is an interconnected system, driven by energy exchange between its parts.

G. A. Folkertsma and S. Stramigioli. *Energy in Robotics*. Foundations and Trends® in Robotics, vol. 6, no. 3, pp. 140–210, 2017.
DOI: 10.1561/2300000038.

1

Introduction

The physical world is governed by energy.
From the kinetic energy in a speeding car to the first law of thermodynamics, energy is the *lingua franca* in all physical domains. It is a coordinate-independent description of the energetic state of a system.

Interactions are almost exclusively[1] characterised by energy exchange.
From a battery, through an electric motor—via the magnetic fields—to the mechanical system of a robot: the power or exchanged energy can be traced across all these physical domains. While a car speeds up because the engine applies a torque on the wheels through a set of transmissions, this effort is really a means of pouring energy from the petrol or battery into the kinetic energy of the car as a moving mass.

Many applications in robotics are concerned with energy: the amount of kinetic energy in the robot (e.g. for safety issues), a periodic motion—oscillation—with a certain amplitude (i.e. total energy), energy-efficiency objectives, and storing and releasing energy in springs for explosive motions are some examples.

[1] Certain interactions, like ideal constraints, can influence motion without energy exchange.

By not solely considering signals, but rather the energy in robotic systems explicitly, more insight can be gained, control problems may become easier and a "feel" for the actual physical processes emerges. This energy-based perspective need not focus on only the control system, nor only on the description of the physical robot. We present a holistic, energy-based view of robotic systems: **Energy in Robotics**. To achieve this holistic view, we shall address the following topics:

1. Energy-based formulation of physical systems: Port-Hamiltonian System theory.
2. Passivity and stability in robotic systems.
3. Measurement and control of energy flowing through actuators.
4. Energy-based controller design: energy shaping and energy routing in the controller.
5. Energy-based system design: shaping the energy flows in a physical robotic system.

The use of energy in robotics is broader than just these topics: there are for example energy-based navigation methods; and in control theory there is a strong link between Lyapunov's stability theorem and energy. The focus of this paper is on the cyber-physical interaction: the study and control of energy flows between the physical system and the controller.

1.1 Port-Hamiltonian systems

Hamiltonian mechanics is a theory of classical mechanics similar to Lagrangian mechanics. The classical canonical formulation is described by a set of equations governing the *Hamiltonian*:

$$\begin{aligned} \frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}t} &= -\frac{\partial \mathcal{H}}{\partial \boldsymbol{q}} \\ \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t} &= +\frac{\partial \mathcal{H}}{\partial \boldsymbol{p}} \\ \mathcal{H} &= T + V. \end{aligned} \tag{1.1}$$

$\mathcal{H}$ is the Hamiltonian, the sum of kinetic T and potential energy V, i.e. the *total internal energy* of the system; $\boldsymbol{q}$ and $\boldsymbol{p}$ are the generalised coordinates and momenta, respectively. A generalised coordinate is e.g. a position, or charge displacement in the electrical domain. Mechanical momentum is e.g. $p = mv$; in the electrical domain it is the state variable of a inductor, the magnetic flux.
Hamiltonian mechanics are suitable for energy-based modelling and control: the total energy $\mathcal{H}$ is expressed explicitly in the equations.

Example 1.1. A simple example of a physical system described with Hamiltonian mechanics is the mass-spring oscillator. The position $\boldsymbol{q}$ is the spring deflection; momentum $\boldsymbol{p}$ is the momentum of the mass, $\boldsymbol{p} = m \cdot v$. With kinetic energy $T = p^2/(2m)$ (mass m) and potential energy $V = q^2/(2C)$ (C is the compliance of the spring, the inverse of its stiffness) the dynamic equations become:

$$\begin{aligned} \mathcal{H} &= \frac{p^2}{2m} + \frac{q^2}{2C} \\ \frac{\mathrm{d}p}{\mathrm{d}t} &= -\frac{q}{C} \\ \frac{\mathrm{d}q}{\mathrm{d}t} &= \frac{p}{m}. \end{aligned} \tag{1.2}$$

Of course, in the equation for p we recognise $\dot{p} = F$, Newton's second law; in this case $m\dot{v} = Kq$. The equation for q is the obvious $\dot{q} = v$. ⟨*example end*⟩

This example shows that energy is explicitly modelled: when solving the equations one will see the energy flow between T and V. In this closed system without friction, the total energy $\mathcal{H}$ is conserved.

In robotics, however, there is always interaction: between mechanical parts, across domains through transducers, and with the environment. For this interaction, the sub-systems must be interconnected. This interconnection can be described by so-called *power ports*: interfaces that transfer energy between elements, domains, systems. A power port is always a pair of variables whose pairing characterises the power exchange, e.g. force and velocity or voltage and current.

In port-Hamiltonian systems theory, a common representation is the causal Poisson framework representation, which is an input-state-output representation. In this representation, all the states like q and p are collected in a single state vector which may even be a combination of generalised moments and displacements and indicated as x:

$$\begin{aligned} \dot{x} &= \big[J(x) - R(x)\big]\frac{\partial \mathcal{H}}{\partial x}(x) + g(x)u \qquad && x \in \mathcal{X}, u \in \mathbb{R}^m \\ y &= g^\top(x)\frac{\partial \mathcal{H}}{\partial x}(x), && y \in \mathbb{R}^m \end{aligned} \tag{1.3}$$

where $J(x) = -J^\top(x)$, $R(x) = R^\top(x) \geq 0$. J is an internal interconnection matrix; R is a resistive structure. g represents the interconnection, and therefore effect, of the port variables on the state variables—and vice versa.
The matrix J is a *power-continuous* interconnection by its skew-symmetry, whereas R models pure resistive losses of the system, as can be seen by taking the time derivative of the Hamiltonian:

$$\begin{aligned} \dot{\mathcal{H}}(x) &= \frac{\partial \mathcal{H}}{\partial x}^\top(x) \cdot \dot{x} \\ &= \frac{\partial \mathcal{H}}{\partial x}^\top(x)\Big[J(x) - R(x)\Big]\frac{\partial \mathcal{H}}{\partial x}(x) + \frac{\partial \mathcal{H}}{\partial x}^\top(x) \cdot g(x)u \\ &= -\frac{\partial \mathcal{H}}{\partial x}^\top(x)R(x)\frac{\partial \mathcal{H}}{\partial x}(x) + y^\top u, \end{aligned} \tag{1.4}$$

which is the power supplied through the port $y^\top u$, minus the power lost to friction, quadratic on $R(x)$.

Example 1.2. Consider the mass-spring-damper system in Figure 1.1: it does not have an external interaction port, so $g(x) \equiv 0$, hence the Hamiltonian should change only with the quadratic $R(x)$ term of (1.4).

The state vector comprises p and q as in Example 1.1; the damping force $F_\mathrm{b} = b \cdot v = b \cdot p/m$ is modelled in the R matrix.

$$\begin{aligned} \mathcal{H}(x) &= \frac{p^2}{2m} + \frac{q^2}{2C} \\ \begin{pmatrix} \dot{p} \\ \dot{q} \end{pmatrix} &= \left[\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} - \begin{pmatrix} b & 0 \\ 0 & 0 \end{pmatrix}\right] \begin{pmatrix} p/m \\ q/C \end{pmatrix} \end{aligned} \tag{1.5}$$

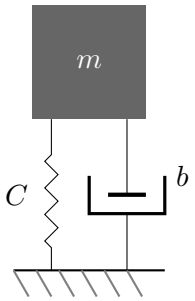


Figure 1.1: A mass-spring-damper system. (Example 1.2)

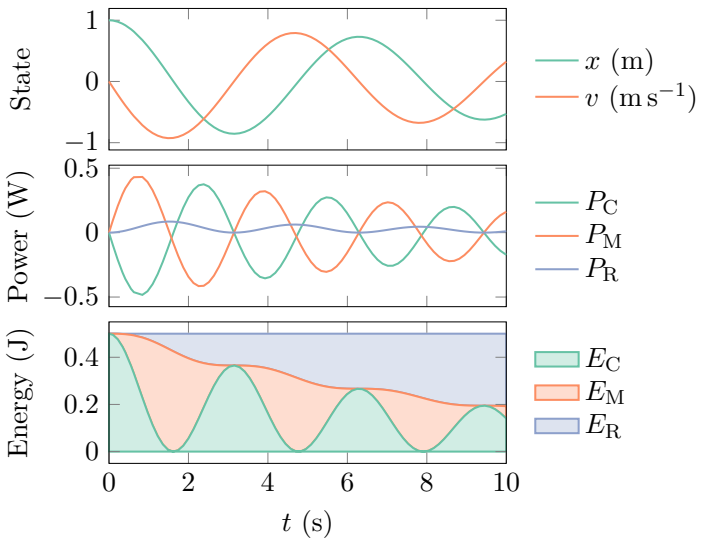


Figure 1.2: Simulation of the mass-spring-damper system of Figure 1.1. Energy flows back and forth between the spring and mass, and is dissipated in the damper. (Example 1.2)

Figure 1.2 shows a simulation of this example system, with $C = 1\,\mathrm{m\,N^{-1}}$, $b = 0.1\,\mathrm{N\,s\,m^{-1}}$, $m = 1\,\mathrm{kg}$, $x(0) = (0 \quad 1\,\mathrm{m})^\top$. Especially the plot of the energy shows how the Hamiltonian $(E_\mathrm{M} + E_\mathrm{C})$ decreases with the energy dissipated in the damper, as expected from (1.4). (E_M and E_C are the first and second term of the Hamiltonian of (1.5); E_R is the energy dissipated by the damper, given by $\int F_\mathrm{b} v \,\mathrm{d}t = \int b v^2 \,\mathrm{d}t$.) ⟨*example end*⟩

Example 1.3. An example of a system with an external port is the sliding mass, with an actuator applying a force on it, as in Figure 1.3. The only state is the momentum p. Choosing F as the input determines $g(x) = 1$ and the dynamic equations are:

$$\begin{aligned} \mathcal{H}(x) &= \frac{p^2}{2m} \\ \dot{x} = \dot{p} &= \big[(0) - (b)\big] \cdot \frac{p}{m} + (1)F \\ y &= (1)^\top \frac{p}{m}. \end{aligned} \tag{1.6}$$

The choice for F as input has made $y = p/m = v$, such that the product of input and output is power and this is indeed a *power port*.

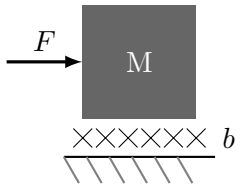


Figure 1.3: A mass sliding on a surface with friction, with a port to the environment: the actuator force. (Example 1.3)

Simulation results of this system (with $m = 1\,\mathrm{kg}$, $b = 0.5\,\mathrm{N\,s\,m^{-1}}$, $F = 0.5\,\mathrm{N}\mathbb{1}(t-1)$) are shown in Figure 1.4. The difference between the power injected by the actuator ($P_\mathrm{F} = v \cdot F$) and the power lost in friction ($P_\mathrm{R} = bv^2$), shaded in the middle graph, is exactly equal to the time derivative of the Hamiltonian, $\dot{E}_\mathrm{M} = P_\mathrm{M}$. ⟨*example end*⟩

Finally, the port of the Port-Hamiltonian System is an interface: the system can be connected to other systems through this power port. The *interconnection* between two or more PHS is described by a Dirac structure, which is a power-continuous coupling of the port variables. In fact, the mass-spring-damper of Example 1.2 can be viewed—and modelled—as three PHS, one for each element, interconnected by a Dirac structure, as in Figure 1.5. The interconnection of Port-Hamiltonian Systems is again a Port-Hamiltonian System, with a Hamiltonian that is the sum of the two systems' Hamiltonians and a new internal interconnection matrix J that incorporates the (old, external) Dirac structure.

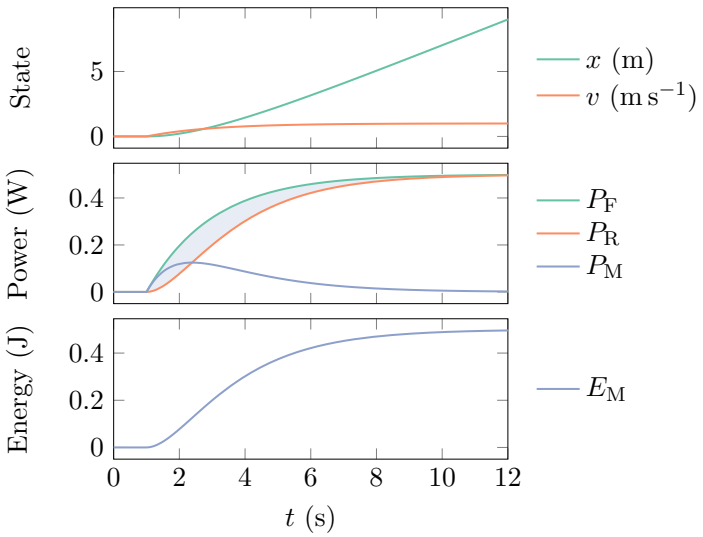


Figure 1.4: Simulation of the sliding mass in Figure 1.3. The difference between the power supplied through the port, P_{F}, and the power lost to friction, P_{R}, is equal to the time derivative of the Hamiltonian E_{M}. (Example 1.3)

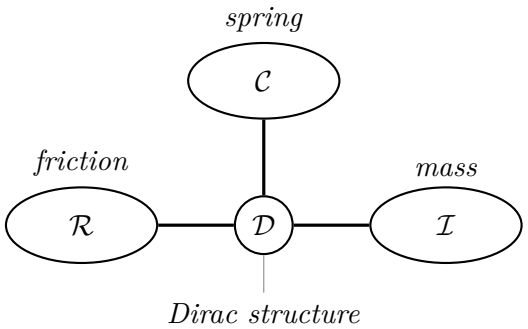


Figure 1.5: A Dirac structure is a power-continuous interconnection between Port-Hamiltonian Systems. This figure shows the system of Figure 1.1 as three interconnected elements, or systems.

An excellent introductory overview of Port-Hamiltonian Systems Theory can be found in van der Schaft and Jeltsema (2014).

2

Energy in controlled physical systems

2.1 Passivity

Willems (1972) introduced the concept of *dissipative dynamical systems* as follows. Consider a dynamical system Σ with state x, input u and output y and a real-valued function $w(u, y)$ called the *supply rate*. If a non-negative *storage function* $S(x)$ can be found such that:

$$S(x_0) + \int_{t_0}^{t_1} w(u(t), y(t))\,\mathrm{d}t \geq S(x_1) \tag{2.1}$$

then the system is said to be *dissipative*.

The supply rate and storage function can be arbitrarily chosen in principle. However, when describing the dynamics of physical systems using Port-Hamiltonian Systems Theory, a natural choice arises: the actual "stored" internal energy $\mathcal{H}(x)$ for $S(x)$ and the energy exchanged through the external port as supply rate; $w(u, y) = y^\top u$ and $S(x) = \mathcal{H}(x)$. As seen before in (1.4):

$$\dot{\mathcal{H}}(x) = y^\top u - \frac{\partial \mathcal{H}}{\partial x}^\top (x)\ R(x)\ \frac{\partial \mathcal{H}}{\partial x}(x). \tag{2.2}$$

This can be recognised as the differential form of (2.1): since $R = R^\top \geq 0$, the square term on the RHS is ≥ 0. The storage function is the energy

present in the system and the supply rate $y^\top u$ is the power transferred to the system through its port.

Physical systems for which the inequality holds are *passive* systems. If $R = 0$, i.e. there is no dissipation[1], then equality holds in (2.2) and the system is said to be *conservative* or *lossless*.

We refer to van der Schaft (2017) for a unified and in-depth treatment of passivity and L_2-gain theory; dissipative systems; nonlinear stability; and other related subjects. In this section we look at the implications of passivity for robot control systems.

2.1.1 Passivity as a must

Many robots are controlled in a non-passive way, for example through PID joint controllers or non-passive state feedback. This leads to good performance, and generally stability can be proven for the free-standing robot. However, as soon as the robot interacts with its environment, things become different—especially if it is an unknown environment. In that case, it turns out that a passively-controlled robot is a necessary condition for stability, as shown in the following theorem, first presented in (Camlibel et al., 2015, Ch 3).

Theorem 2.1. Given any *non-passive* system Σ with input-output pair (u, y), then there always exists a *passive* system $\tilde{\Sigma}$ that, when connected to Σ, will give rise to unbounded behaviour of the interconnection of Σ and $\tilde{\Sigma}$.

Proof. Take the Hamiltonian of Σ as storage function and the natural supply rate $y^\top u$. By the definition of passivity, (2.1), non-passiveness of Σ implies that $\exists\ \bar{u}(t)$ such that the integral of the supply rate is unbounded, which means we can extract infinite energy from the system. Indicate with $\bar{y}(t)$ the output corresponding to the input $\bar{u}(t)$. This means that we can define the extracted energy function $H_o(t)$ as:

$$H_o(t) = \int_0^t \bar{u}(s) \cdot \bar{y}(s)\, \mathrm{d}s. \tag{2.3}$$

[1] Properly speaking, energy cannot be dissipated for the first principle of thermodynamics, but what is meant here is what is called free energy and this means irreversible transformation of energy to heat.

By construction, $\lim_{t\to\infty} H_o(t) = \infty$. This implies that due to the continuity of $H_o(t)$, $\exists$ a bounded $H_{\min} := \min_t H_o(t)$

We will now constructively define a passive system $\tilde{\Sigma}$ that will generate the input $\bar{u}(t)$ as its output $\tilde{y}$:

$$\dot{x} = n(t)\tilde{u} \tag{2.4}$$

$$\tilde{y} = n(t)\frac{\partial \tilde{H}}{\partial x}, \tag{2.5}$$

with $\tilde{H}(x) = \frac{1}{2}x^2$ and $n(t) = \frac{\bar{u}(t)}{\frac{\partial \tilde{H}}{\partial x}}$. It is easy to see that this system is passive (even conservative) with storage function $\tilde{H}(x)$. By initialising $x(0) = \sqrt{2H_{\min} + \epsilon}$ for any $\epsilon > 0$, it can be seen that by construction $\frac{\partial \tilde{H}}{\partial x}(t) > 0 \ \ \forall t > 0$. Therefore, it is always possible to calculate $n(t)$. By setting as interconnection $\tilde{u} = y$ and $u = \tilde{y}$ $(= \bar{u})$, we by construction have that

$$\lim_{t\to\infty} H_0(x) = \lim_{t\to\infty} \tilde{H}(x) = \infty \Rightarrow x \to \infty, \tag{2.6}$$

which proves divergence of the coupled system (and thus instability for any equilibrium), due to its diverging state x. □

It is worth thinking about this relatively simple proof for a while, because **the implications of the theorem are far-reaching**. The theorem applies to any (linear and non-linear) system. This means that, if a controlled robot is not passive, it is possible to construct an environment—possibly by a second controlled robot— that is passive, but when connected to the original robot would result in an energetically unbounded system.[2]

A corollary to the above: if a controlled system *is* passive, it is *guaranteed* to be stable. This is a strong argument to create a passive behaviour for any robot that will potentially interact with an unknown environment, in order to ensure **bounded and safe behaviour**.

One way to obtain this stability and safety guarantee is by using an energy monitor, a supervisor that keeps the energy injected by actuators in check. A different, model-independent way is presented in §2.3.

[2]Although the constructed system *might* not be encountered in practice, the theorem clearly shows that from a control perspective, passivity is indeed a necessary condition.

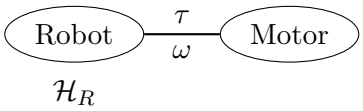


Figure 2.1: A robot as a Port-Hamiltonian System connected to its actuator through a power port. The rate of change of its Hamiltonian $\mathcal{H}_\mathrm{R}$ is the supplied power, $\tau \cdot \omega$.

2.2 Energy and distributed architectures

If a robotic system is seen as a Port-Hamiltonian System, then an actuator interacts with it through a port. In the case of an electric motor, the port variables are torque τ and rotational velocity ω, Figure 2.1. Not regarding internal losses and interaction with an environment, the change of the robot's internal energy, $\dot{\mathcal{H}}_\mathrm{R}$, is the supplied power, $\tau \cdot \omega$.

2.2.1 Measuring energy

In order to measure the energy supplied to the robot, one could measure the electrical power going into the motor. However, an—often large—part of that energy is directly converted into heat, rather than mechanical energy. Instead, it is possible to directly and precisely measure the supplied mechanical energy ΔE between two times t_0 and t_1 (2.7).

Electrical motors are usually current-controlled, at least in an inner loop, meaning that their torque τ is precisely known if the motor constant K_m is known (2.8). Moreover, most digital control systems use a zero-order-hold (ZOH) for their outputs during a sample period. Suppose that t_0 and t_1 are two consecutive instants in which the sampling of the position of the actuated joint takes place. At the same time the new set-point of a current, and therefore a torque, takes place and it will be held constant (ZOH) until the next sampling instant. In this way, the current is kept constant over the integration period (2.9) and the energy injected by the actuator can be measured exactly (2.10). In this last equation, $q(t_i)$ is a position measurement of the motor at a sampled time instant.

$$\Delta E = \int_{t_0}^{t_1} \tau(t) \cdot \omega(t)\,\mathrm{d}t \qquad \text{supply rate} \qquad (2.7)$$

$$= \int_{t_0}^{t_1} K_m i(t) \cdot \omega(t)\,\mathrm{d}t \qquad \text{motor constant} \qquad (2.8)$$

$$= K_m i(t_0) \int_{t_0}^{t_1} \omega(t)\,\mathrm{d}t \qquad \text{ZOH} \qquad (2.9)$$

$$= K_m i(t_0) \cdot (q(t_1) - q(t_0))\,. \qquad (2.10)$$

It is important to realise that this is indeed the *exact* amount of injected energy, not an estimate. It can be measured if two conditions are met:

1. A zero-order hold is used for the current or motor torque.
2. A position sensor is collocated with the motor.

Note that there are *no conditions on the sample time*. This method of *passive sampling* was first introduced in Stramigioli et al. (2002). Later, in Stramigioli et al. (2005), the sampled energy was sent as an "energy packet" through discrete-time scattering between two haptic devices, ensuring a passive interconnection regardless of communication delays or losses.

If there is uncertainty in the measurement of i or q, that uncertainty will carry to ΔE. Specifically, in the case of angular encoders with a resolution of Δq, the maximum error of the measurement $\Delta \hat{E}$ is known:

$$q(t_i) \in \left[\hat{q}(t_i) - \frac{\Delta q}{2}, \hat{q}(t_i) + \frac{\Delta q}{2}\right) \Rightarrow |\Delta \hat{E} - \Delta E| < K_m i(t_0) \Delta q. \quad (2.11)$$

In this equation, $\hat{q}$ is the quantised measurement of q.

Example 2.1. Consider the system in Figure 2.2: a mass can be lifted by an electric motor, through a gearbox and pulley. The motor current is set by a PID controller. This system is simulated, where the setpoint is a cycloid (smooth step) from 0 m to 0.5 m for the mass, executed between 1 s and 3 s. See Table 2.1 for all parameters of the simulation.

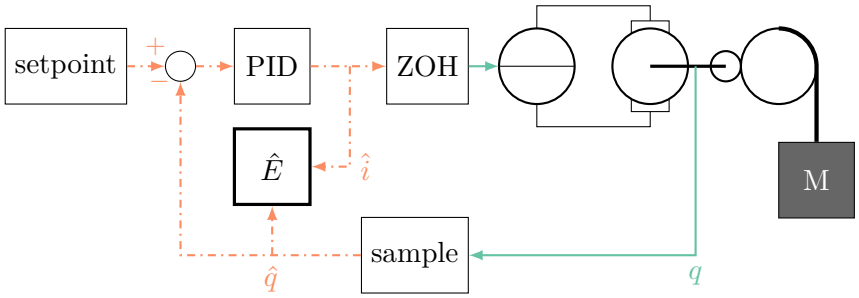


Figure 2.2: A mass that can be lifted by a current-controlled motor. A discrete-time PID controller tries to follow a cycloid setpoint from 0 m to 0.5 m between 1 s and 3 s. The energy injected by the motor is measured exactly by the block $\hat{E}$.

Parameter	Value	Description
K_m	$0.04\,\mathrm{N\,m\,A^{-1}}$	Motor constant
J_m	$1 \times 10^{-8}\,\mathrm{kg\,m^2}$	Motor inertia
R	$4 \times 10^{-7}\,\mathrm{N\,s\,rad^{-1}}$	Viscous motor friction
n	30	Gear reduction ratio
r_p	0.02 m	Pulley radius
M	1 kg	Mass
K_P	1.5×10^{-4}	PID proportional gain
τ_D	0.4 s	PID derivative time constant
τ_I	5 s	PID integral time constant

Table 2.1: Parameters used in the simulation of Figure 2.2.

The energy measurement block $\hat{E}$ implements the result of (2.10) as follows, where $\hat{i}$ and $\hat{q}$ indicate the discrete-time variables, as labelled in Figure 2.2.

procedure INITIALISATION
  $E \leftarrow 0$
  $q_\mathrm{prev} \leftarrow 0$
  $i_\mathrm{prev} \leftarrow 0$
end procedure
procedure UPDATE

$E \leftarrow E + K_m \cdot i_{\text{prev}} \cdot (\hat{q} - q_{\text{prev}})$
$i_{\text{prev}} \leftarrow \hat{i}$
$q_{\text{prev}} \leftarrow \hat{q}$
end procedure

The result of the simulation with various sample frequencies f_s for the discrete-time part is shown in Figure 2.3. While the controller performance clearly deteriorates for very low sampling frequencies, the energy measurement is actually always exactly equal to true injected energy.

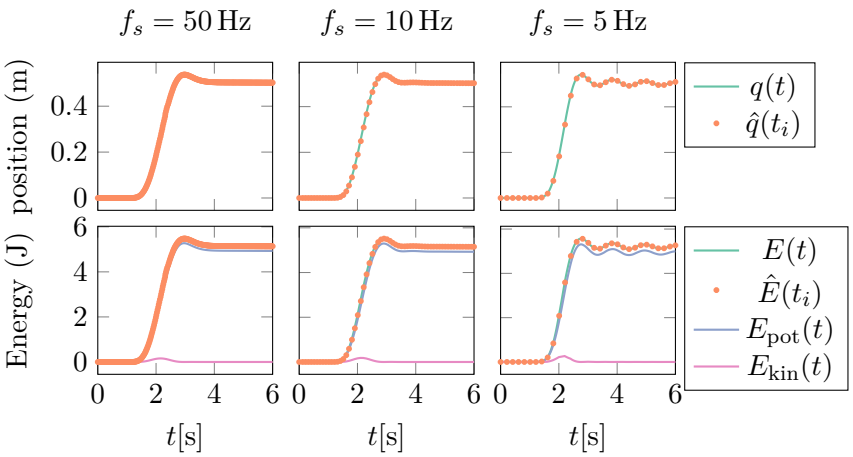


Figure 2.3: Simulation of a PID-controlled mass that is lifted to 0.5 m. The energy is measured exactly, according to (2.10). Most of the injected energy E goes into potential energy of the mass E_{pot}, but some is lost to friction. While moving, the mass has some kinetic energy E_{kin}. Note that for low sample frequency the controller loses performance, but the energy is still measured exactly.

⟨*example end*⟩

Example 2.2. When the position measurement is quantised, for example by an encoder with limited counts per revolution (cpr), the energy bounds calculation from (2.11) can be implemented as follows:

procedure INITIALISATION
$E_{\text{min}} \leftarrow 0$
$E_{\text{max}} \leftarrow 0$

$q_{\text{prev}} \leftarrow 0$
$i_{\text{prev}} \leftarrow 0$
end procedure
procedure UPDATE
$q_{\text{diff}} \leftarrow \hat{q} - q_{\text{prev}}$
if $q_{\text{diff}} > 0$ **then**
$E_{\min} \leftarrow E_{\min} + K_m i_{\text{prev}} \left(q_{\text{diff}} - 0.5\Delta q\right)$
$E_{\max} \leftarrow E_{\max} + K_m i_{\text{prev}} \left(q_{\text{diff}} + 0.5\Delta q\right)$
else
$E_{\min} \leftarrow E_{\min} + K_m i_{\text{prev}} \left(q_{\text{diff}} + 0.5\Delta q\right)$
$E_{\max} \leftarrow E_{\max} + K_m i_{\text{prev}} \left(q_{\text{diff}} - 0.5\Delta q\right)$
end if
$i_{\text{prev}} \leftarrow \hat{i}$
$q_{\text{prev}} \leftarrow \hat{q}$
end procedure

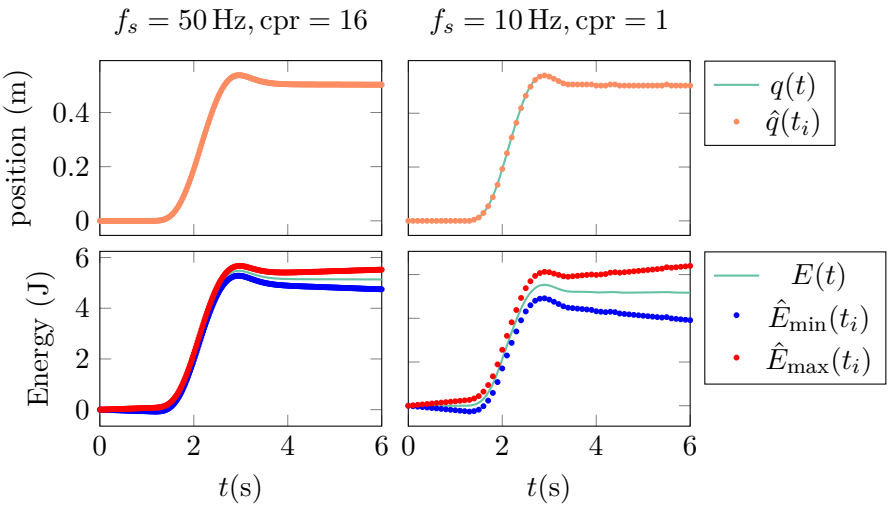


Figure 2.4: Simulation of Figure 2.2, but now with a quantised motor position measurement. Using (2.11), exact bounds on E are calculated. Note that even with only 1 count per revolution (cpr) and low sample frequency, the bounds remain quite close to the actual energy.

In the simulation, the position measurement is quantised by truncation:

$$\hat{q} = q - (q \mod \Delta q); \quad \Delta q = \frac{2\pi}{\text{cpr}}. \tag{2.12}$$

The two simulations in Figure 2.4 show that the true injected energy always lies between the calculated bounds. Considering the second experiment, with a sample frequency of only 10 Hz and an encoder with *only one count per revolution*, this is remarkable indeed.

A potential problem with the method does become apparent from these figures: the bounds are widening as time passes, because theoretically the motors could always be jittering back and forth within a single encoder step in the worst possible way, continuously extracting energy from or injecting energy into the system. In practice, the encoder resolution will be much higher than in the example and this problem mitigated tremendously. Additionally, an external energy observer might be added to compensate for the drift of these bounds. However, the guarantee that the true energy lies strictly within the bounds would be lost. Alternatively, a worst case approach could be used in order to ensure that the energy injected in the system will be overestimated, as in Secchi et al. (2003). ⟨*example end*⟩

We have carried out experiments with a simple 1-DoF robot arm, measuring the energy according to (2.11). When the motor is controlled with a proportional position controller, it behaves as a virtual spring with stiffness K_P (the proportional gain). The "virtual energy" that is inside this virtual spring is given by

$$E_\text{C} = K_\text{P} e^2, \tag{2.13}$$

where e is the position error, or indeed the virtual spring state. This energy should match exactly with the energy that is extracted from the system, i.e. $E_\text{C} = -\hat{E}$. As Figure 2.5 shows, when manually pushing the arm away from its setpoint, the internal controller energy is *exactly equal* to the measured energy.

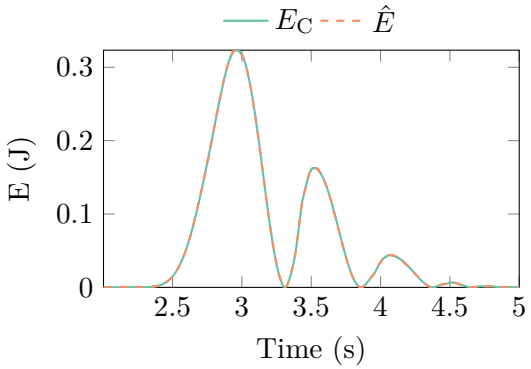


Figure 2.5: Experiment with P-controller on a 1-DoF robot arm. The virtual energy present in the controller, $E_C = K_P e^2$, is exactly equal to the sampled energy $\hat{E}$.

2.3 Energy budgets

From the stability-passivity criterion in §2.1.1, it is clear that in some cases—robots interacting with unknown environments—it is indispensable for stability to have a strictly passive control system. In other cases, it may still be advisable to limit the amount of energy that can be injected into the robot by the actuators: considering that damage and injury strongly correlate with energy transfer, limited energy greatly enhances the safety of a robotic system.

This can be done with energy observers that make use of knowledge of the system, for example by limiting the velocity or force of the motors, as in Tadele et al. (2014). However, with the exact energy measurement scheme, it is possible to limit the energy that actually enters the system, which is a safety measure that is completely independent from the system model (and thus from modelling errors). The first work introducing this concept can be found in Duindam and Stramigioli (2004).

The controller can be given an *energy budget* that is put into a virtual *energy tank*. The energy flowing through the actuators, measured using (2.10), has to come from this tank: ΔE is subtracted from the initial budget. In case the actuators extract energy from the system, for example when they slow down a mass, the energy flows back into the tank (since ΔE of (2.10) is negative). When the tank is empty,

the controller is no longer allowed to inject additional energy into the system: then the controller is guaranteed to be passive. See Figure 2.6 for a figure illustrating the concept.

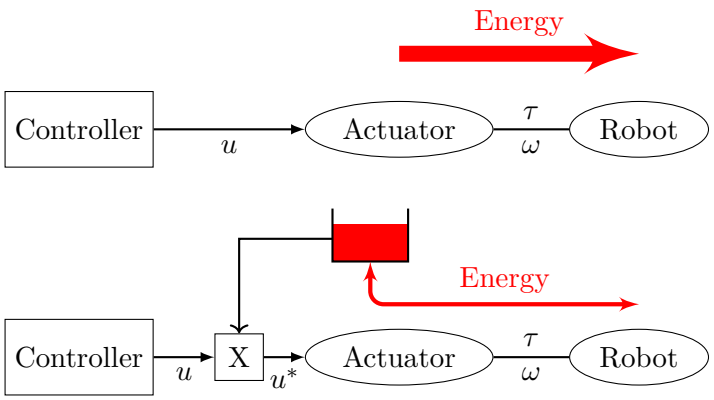


Figure 2.6: Actuators can inject unlimited amounts of energy into the system if no special care is taken. For safety, passivity and stability, the robot system can be equipped with a virtual energy tank from which all energy that goes into the system must come. Any controller can be augmented with this safety/passivity layer. The "X" block modulates the actuator signal as described in Figure 2.7.

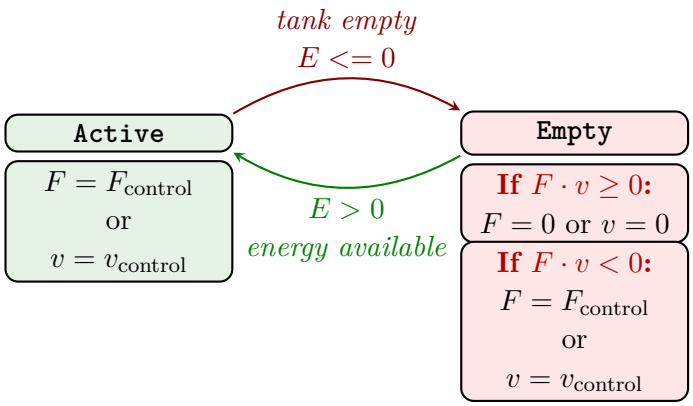


Figure 2.7: A Finite State Machine implementation that, given the current energy tank level E, prevents further energy injection if the tank is empty.

Since the power flowing into the robot is given by $F \cdot v$ (or $\tau \cdot \omega$), preventing the injection of more energy is achieved by forcing either

the force or the velocity to zero. (Note that a control action that would cause energy *extraction* can still be allowed.) See Figure 2.7 for a Finite State Machine implementation of the energy tank.

Example 2.3. Consider the system depicted in Figure 2.8: a mass is lifted by a pulley, driven by an electric motor. The motor command is a constant torque resulting in a force on the mass greater than its weight, so the bucket is lifted. This action injects much (potential) energy into the system. By implementing the energy-sampling algorithm of (2.10) and the FSM of Figure 2.7, the amount of injected energy is limited: the actuator is given an energy budget of 5 J; the mass of 1 kg is lifted until this energy has been used up, after which the mass is held steady. The experiment and corresponding simulation are shown in Figure 2.9.

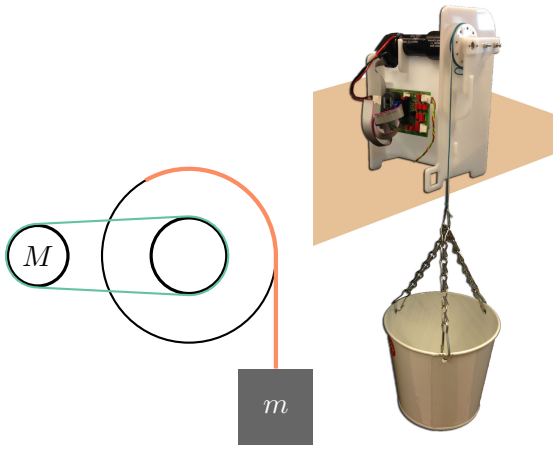


Figure 2.8: A motor M that lifts a mass m by means of a pulley.

Around $t = 6.5\,\mathrm{s}$, the mass is pulled down manually, which—acting against the holding torque—injects energy back into the virtual tank, allowing the controller to resume normal operation until the energy is used up again.

Note that the mass does not quite reach its expected height of $h = 5\,\mathrm{J}/(1\,\mathrm{kg}\cdot 9.8\,\mathrm{m\,s^{-2}}) \approx 0.5\,\mathrm{m}$ due to friction in the gearbox between motor and pulley. Dissipation can be a serious problem in passivity-based control, but as shown in Theorem 2.1, non-passive control may cause serious stability and safety problems. ⟨*example end*⟩

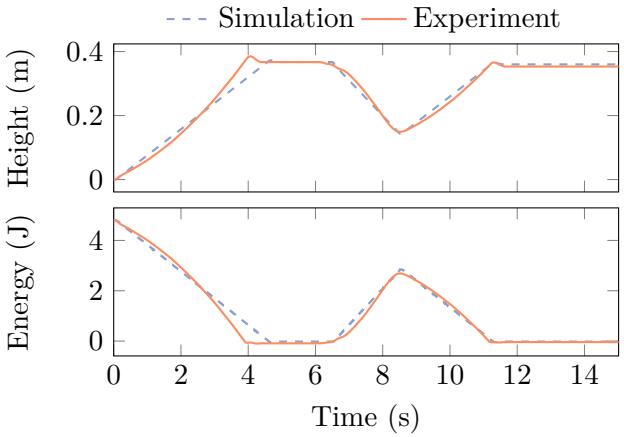


Figure 2.9: Lift experiment and simulation: the mass is lifted by a constant torque, with the energy tank algorithms in place. Around $t = 6.5\,\mathrm{s}$, the mass is pulled down manually.

With the energy tank and an energy budget, the controller is guaranteed to be passive, regardless of the control algorithms. To see this, take the integral form of the dissipation inequality (2.1):

$$S(x_0) + \int_{t_0}^{t_1} w(u(t), y(t))\,\mathrm{d}t \geq S(x_1). \tag{2.14}$$

With $S(x_0)$ the initial energy in the tank and supply rate $w = \tau \cdot \omega$, it is clear that the energy left in the tank at time t_1, $S(x_1)$, is the initial energy $S(x_0)$ minus the injected energy, satisfying the (in)equality.

Hence, with a passive mechanical system driven by this *passivated* controller, the overall system is also guaranteed to be passive and thus stable (energetically bounded, Theorem 2.1). It is important to realise that the control modulation technique of Example 2.3 probably leads to a discontinuous control signal, which may be undesirable. Furthermore, strict passivity in systems with lots of friction may lead to unacceptable performance loss in other areas, such as position accuracy or settling time. In these cases, a supervising energy observer may be used to inject extra energy in the system in a controlled fashion.

3

Control by interconnection

In classical control theory, the inputs and outputs of the controller and plant are treated as signals. A sensor in the plant gives feedback to the controller (y); the output of the controller is an input to the plant (u) (Figure 3.1). The dimensions of u and y do not necessarily match in a Multi-Input, Multi-Output (MIMO) control system.

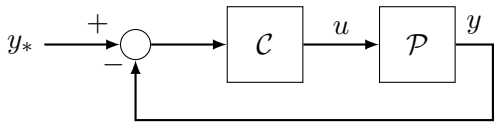


Figure 3.1: A classical control system treats the inputs and outputs of plants and controller as signals.

This method of control does not lend itself well for energy-based robotics: if the sensors are not necessarily collocated with the actuators, it is impossible to measure or influence the exchanged energy as in §2.2. (The first requirement for *exact* energy measurement was collocated sensors and actuators.) Moreover, it is often hard to give a physical interpretation of a signal-based control law, which makes the controller itself sometimes difficult to understand.

From Figure 2.1 it is clear that all actuators in a robotic system actually exchange energy and physical quantities in a bi-directional way: in the case of electrical motors, electrical power is converted to mechanical power and exchanged with the system. This bi-directional flow of energy and physical variables can be extended to the control system: with this paradigm, the controller is no longer a signal-processor with separate inputs and outputs, but is a (Port Hamiltonian) system connected to the robot via power ports, as shown on the left in Figure 3.2.

In reality, the actuators sit between the controller and the plant. Or, drawing the boundary somewhere else, the actuators may be considered part of the plant, too, and the electrical circuit (current amplifiers) sit between controller algorithm and mechanical plant. In any case, the actuators are connected to a power source and have losses, so the drawing on the right in Figure 3.2 is more accurate. However, with proper considerations as described in the previous section, §2.2, the actuator does become a power-continuous "transparent" connection between controller and plant.

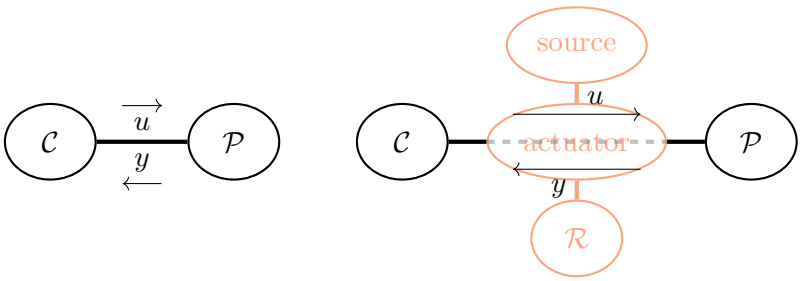


Figure 3.2: The controller is a (Port-Hamiltonian) system connected to the robot via power ports (left). In real robotic systems, the actuators sit between the controller and mechanical system, but can be made "transparent" with the energy sampling method presented earlier.

With both the controller and plant described as a Port-Hamiltonian System, physical interpretation can be given to both of them. And, because the interconnection of two PHSs is again a PHS, this approach is

inherently modular.[1] Most importantly, because the connection between controller and plant is power-continuous, there is direct control over the energy in the system. Passivity or energy-budgeted control are now inherent in the controller design. This approach of control is called *control by interconnection*; see Stramigioli et al. (1998), for likely the first real example in this context; and Ortega et al. (2001) and Ortega et al. (2008) for an in-depth mathematical treatment. In this section, we will consider the application to robotics.

3.1 Impedance control

Impedance control was introduced by Hogan (1985) as an approach to manipulation, after realising that force or position control is inadequate for real interaction tasks:

> "Because of dynamic interaction [with the environment], the manipulator may no longer be treated ... as an isolated system. Strategies directed toward the control [of] position, velocity or force will be inadequate as they are insufficient to control the *mechanical work exchanged* between the manipulator and its environment."

In his three-part paper series, Hogan argues that most environments behave like an admittance—mass-like—so that the controller must behave as an impedance. Often, the controller (actuator) is directly connected to the robot's mass, an admittance, so even without regarding the environment, the controller should be an impedance. This means that a controller is defined by its port behaviour, by the relation between force and velocity. The impedance versus admittance is a "causality" argument, which regards the robot's velocity as an input to the controller and, consequently, the output of the controller is a force. It does not in principle say anything about *how* this relation should be implemented.

In the examples of Hogan, a second order behaviour has been used as an example of such a control strategy. By shaping the second-order

[1] The Port-Controlled Hamiltonian Systems structure is not strictly necessary: it is the power-continuous interconnection that is vital; passivity properties can always be restored with the energy tank approach.

behaviour, the compliance, damping and the felt inertia—also called driving-point inertia—may be changed. Often, this specific implementation and choice has been considered in literature as "*The* impedance control approach," instead of the much more general causality and port-behaviour argument mentioned before. A characteristic example of this debate was published in Volpe and Khosla (1995) and the reaction by Won et al. (1997).

A simpler implementation of impedance control not trying to change the driving point inertia is compliance control. In this approach, a mechanical impedance can be a combination of a spring and a damper, so with x the end-effector position and v its velocity, the control force is determined by:

$$F_{\text{imp}} = -K \cdot (x - x_0) - b \cdot v, \tag{3.1}$$

where x_0 is the end-effector position at zero potential energy and K, b are the controller parameters: stiffness and damping coefficients, respectively. Since the two behaviours that are implemented—the spring and damper—are both physical systems, it is straightforward to describe (3.1) as a Port-Hamiltonian System, and recognise that the combination (F_{imp}, v) is a power port interconnection between the PHS of the controller and that of the physical robot, as depicted in Figure 3.3. The PHS equations of the controller are now:

$$\begin{aligned} \mathcal{H}(x) &= \frac{K}{2}(x - x_0)^2 \\ \dot{x} &= (0)\frac{\partial \mathcal{H}}{\partial x}(x) + (1) \cdot v \\ F &= (1)\frac{\partial \mathcal{H}}{\partial x}(x) + bv. \end{aligned} \tag{3.2}$$

(The term bv is a direct feed-through from input to output.) The power-continuous interconnection is characterised by the Dirac structure or network structure between the robot (R) and controller (C) as

$$F_{\text{R}} = -F_{\text{C}}, \quad v_{\text{C}} = v_{\text{R}}. \tag{3.3}$$

The power-continuity of this interconnection can be shown by considering the power flowing through both ports:

$$P_{\text{R}} = F_{\text{R}} v_{\text{R}} = -F_{\text{C}} v_{\text{R}} = -F_{\text{C}} v_{\text{C}} = -P_{\text{C}}. \tag{3.4}$$

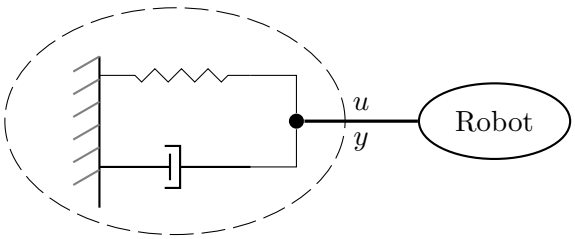


Figure 3.3: Port-Hamiltonian interpretation of the impedance control strategy: the port variables u and y are the controller force and velocity, respectively.

The sum of P_{R} and P_{C} is zero and the network structure is indeed a proper Dirac structure. It is actually possible to expand the interconnection of this controller PHS with the robot as a mass into the same system as Figure 1.5, where the mass $\mathcal{I}$ is "physical" and the spring and friction are "virtual", implemented in the controller.

A last important issue which should be considered is the fact that the interconnection is defined via port variables, which are force and velocities in the mechanical domain. By implementing the compliance, the *position* of the robot is used instead. A proper formulation to understand how to handle this issue is proposed in Stramigioli et al. (1998).

3.1.1 Port behaviour and interaction

It is important to realise a subtle but fundamental issue which is often not regarded in the literature. Force control and position control are not always well-posed problems, considering that the force or position of the end effector of the robot is a consequence of both the controller—which we *can* influence—and of the environment—which we *cannot* influence. To clarify the issue: it is impossible to apply a desired force if no environment is touched, or to position an end-effector behind a rigid wall.

On the other hand, it is always possible and well-defined to achieve the dynamics of (3.1) and controlling K, x_0 and b, independently of whether we move in free air or touch a very stiff wall. It therefore makes sense to control the behaviour (K, x_0, b): this is possible in all

interaction situations, unlike when trying to control signals like force or position (Stramigioli et al. (2002)). This issue will be treated in more detail hereafter.

3.1.2 Stability of impedance control

An important note on stability and passivity with impedance control: as drawn and analysed here, the reference position x_0 is assumed to be constant. The virtual spring and damper are connected to a fixed world, the controller PHS is a passive physical system, hence the total system is guaranteed to be stable at the energetic minimum.

Often, however, x_0 is a time-dependent reference trajectory. In this case, although the controller *seems* to be a passive dynamical system, it is in fact in general not passive and therefore no guarantees can be given on stability. This is because the controller PHS has a second input v_0 ($\dot{x}_0$), through which energy can be injected into the spring-damper and subsequently into the robot.

3.1.3 Comparison to position and force control

Position control Control strategies that aim for a position error $e \to 0$ as $t \to \infty$ are applicable to robotic manipulators that do not touch a (rigid) environment—whose setpoint does at least not move into the solid environment—and are thus very different from impedance-controlled robots. However, there is a large overlap as well: a PD-controller is identical to a spring-damper impedance controller where K_P and K_D (the proportional and derivative gain) are equal to K and b of (3.1).

The distinction lies in the control objective: an impedance controller aims to implement a certain stiffness and damping, whereas in position control the gains are usually maximised to reach a minimal position error. In the case of integrating action, the position controller loses all direct equivalence with impedance control. The equivalence *could* be regained by varying the rest position of the spring as derived below in (3.5)—but as remarked before, varying x_0 generally injects energy, hence passivity is lost.

$$F_{\mathrm{PI}} = Ke + K_{\mathrm{I}} \int e \,\mathrm{d}t \qquad e = x_s - x \tag{3.5}$$

$$\begin{aligned} F_{\mathrm{imp,K}} &= K(x_0 - x) \qquad & x_0 &:= x_s + \frac{K_{\mathrm{I}}}{K} \int (x_s - x) \,\mathrm{d}t \\ &= K\left(x_s + \frac{K_{\mathrm{I}}}{K} \int (x_s - x) \,\mathrm{d}t - x\right) \\ &= Ke + K_{\mathrm{I}} \int e \,\mathrm{d}t. \end{aligned}$$

Force control In interaction tasks, one can also control the force applied to the environment. The main drawback is that force control is only possible if there is contact between the manipulator and the environment. (If not touching anything, there is nothing to apply a force *on*.) In some cases there *is* an equivalence between force control and impedance control: if the manipulator does not move—for instance with an infinitely stiff environment—then $\dot{x} = 0$ in (3.2) and F is constant; Volpe and Khosla (1995). However, there is again a clear distinction: impedance control determines the dynamic interaction behaviour of the manipulator and the observed equivalence only holds in some very specific cases; Won et al. (1997).

Impedance control In specific situations there can be an equivalence to some form of position control; and in other cases to some form of force control. However, impedance control is more general in that it determines the *port-behaviour* at the *interaction port* between robot and environment. That behaviour is always independent of the specific environment and can be controlled independently from it.

Example 3.1. Implementing the spring-damper impedance control of Figure 3.3 in practice leads to two difficulties:

1. Real motors are usually equipped with position sensors, not velocity sensors, so implementing the damper requires some form of velocity observer.

2. The actuator limits may cause the actual control force u to deviate from the desired force.

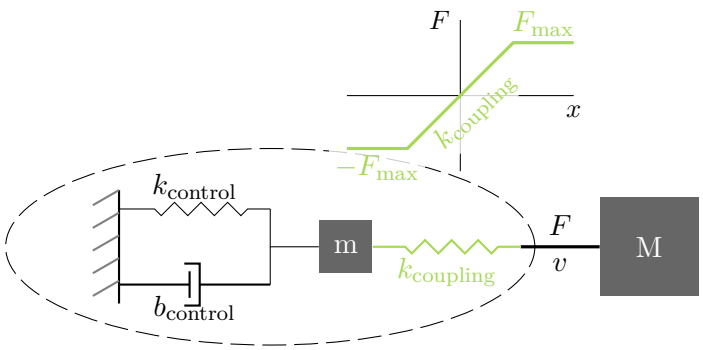


Figure 3.4: Damping injection by extending the regular spring-damper impedance controller with an extra mass and (nonlinear) spring. They solve the problems of velocity observation and actuator saturation.

Both these issues can be solved in an elegant way by extending the controller with an extra (virtual) spring and mass, as shown in Figure 3.4 and introduced in Stramigioli (1996); Ortega et al. (1994).

1. If the coupling spring is sufficiently stiff and the virtual mass sufficiently small, that is to say,

$$k_{\text{coupling}} \gg k_{\text{control}}; \quad \text{m} \ll \text{M}, \tag{3.6}$$

then $v_{\text{m}} \approx v_{\text{M}}$. Because the virtual mass m is simulated, its velocity is known; and to calculate the control force u, only the *displacement* of M needs to be known—which is measured by a position sensor.

2. The coupling spring is piecewise linear, flattening at the maximum actuator force $F_{\max}$:

$$F_{\text{coupling}} = \begin{cases} F_{\max} & k_{\text{coupling}} x_{\text{coupling}} \geq F_{\max} \\ -F_{\max} & k_{\text{coupling}} x_{\text{coupling}} \leq -F_{\max} \\ k_{\text{coupling}} x_{\text{coupling}} & \text{elsewhere.} \end{cases} \tag{3.7}$$

This way, the force calculated by the impedance controller can always be met by the actuator and the control system behaves as expected.

⟨*example end*⟩

3.2 Energy shaping

The impedance control law of (3.1) implements a spring pulling the end-effector to the virtual position x_0. Another way to look at this is that the energy function of the PHS of this same controller, (3.2), implements a potential energy function quadratic in x, with a minimum at x_0.

This idea, *energy shaping*, can be applied in a more general way. The method is well-known in the literature; here we look at it from a Port-Hamiltonian System point of view. Consider a robot mechanism described by

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + \frac{\partial V(q)}{\partial q} = \tau, \tag{3.8}$$

where q are the joint positions and τ the applied joint torques; M its inertia matrix; C describes the Coriolis and centrifugal forces; and V the potential energy.

In this fully-actuated collocated control system, it is possible to construct the control force τ from the derivative of some controller potential energy V_c:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + \frac{\partial V(q)}{\partial q} = -\frac{\partial V_\mathrm{c}(q)}{\partial q}, \tag{3.9}$$

such that the closed-loop dynamics become

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + \frac{\partial \left(V(q) + V_\mathrm{c}(q)\right)}{\partial q} = 0. \tag{3.10}$$

In other words, the robot behaves as a mechanism with a new potential energy $V + V_\mathrm{c}$. If $V(q)$ is known, it can even be fully compensated by setting $V_\mathrm{c} = -V + V_\mathrm{c}'$ and the robot given an arbitrary desired potential energy V_c' with its minimum at a desired q_0.

From a Port-Hamiltonian System point of view, the controller is interconnected to the robot's power port $(\dot{q}, \tau)$. The controller's PHS-equations become:

$$\begin{aligned} \mathcal{H}(x) &= -V(x) + V_\mathrm{c}'(x) \\ \dot{x} &= u \\ y &= \frac{\partial \mathcal{H}}{\partial x}(x) \end{aligned} \tag{3.11}$$

with the Dirac structure ($u = \dot{q}, \tau = -y$). These interconnected systems are again a PHS with total energy

$$\mathcal{H}_{\text{tot}}(q, x) = (V(q) - V(x)) + V_{\text{c}}(x) + \dot{q}^\top M(q)\dot{q}. \tag{3.12}$$

Due to the interconnection, $x = q$ and so V_{c} can be designed such that its minimum is at the desired equilibrium q_0.

If there is no or insufficient damping present in the robot, this may be added, e.g. by a damping matrix $B(q) \geq 0$ adding a damping force $B(q)\dot{q}$ to τ of (3.8). It is implemented as a direct feed-through term like in (3.2), by adding a term $B(x)u$ to the output expression of (3.11). This *damping injection* dissipates (free) energy, reducing $\mathcal{H}(x)$ to its minimum of $\mathcal{H}_{\text{tot}}(q_0, q_0)$ sooner, as shown by the dissipation relation of (1.4):

$$\dot{\mathcal{H}}_{\text{tot}}(q, x) = -\dot{q}^\top B(x)\dot{q}. \tag{3.13}$$

3.2.1 Remarks on energy shaping

- In the example above, we have made x equal to q by setting $g(x)$, the input/state/output mapping of (1.3), equal to 1. Instead, $g(x)$ can be used for a coordinate transformation to shape the potential energy in some other coordinate system, e.g. Cartesian end-effector coordinates.

- In the case of underactuated robots, the potential energy may not be fully reshaped. In that case, $V(q) - V(x)$ in (3.12) does not disappear—still, the system converges to the minimum of $\mathcal{H}_{\text{tot}}$. Closely related to this is the issue of state detectability: if some state variables of the plant are undetectable, it is impossible to shape the energy related to those coordinates.

- In general, there might not be a direct one-to-one connection between the full plant state x and the controller state q as assumed in the example. A method to obtain a connection between x and q is to generate *Casimir functions* for the connection (Stramigioli et al. (1998)). Casimirs are conserved quantities of q and x that can be used in Lyapunov candidates to find a stabilising control

law. We refer to Cervera et al. (2007) and van der Schaft and Jeltsema (2014) for more on this topic.

- Energy shaping is a form of passivity-based-control: the controller is a passive Port-Hamiltonian System, so it can inject only a finite amount of energy, namely $\mathcal{H}(x(0)) - \min \mathcal{H}(x)$. This means it is impossible to steer the system to a state in which it dissipates energy, e.g. to a constant nonzero velocity with friction loss. In the presence of dissipation—that is to say, in all relevant practical cases—it is therefore only possible to reach a state $(q, \dot{q}) = (q_0, 0)$. In position control, this is exactly the goal, so this *dissipation obstacle* is of no concern. However, it should be noted that this energy shaping is actually only *potential* energy shaping. If some kind of (periodic) motion is the control objective, also the *kinetic* energy must be controlled, which is addressed in the next section.

3.3 Energy routing

The concept of controlling the energy of a robotic system can be taken further. Before, the (potential) energy of the controller was shaped, thereby shaping the energy of the interconnected system. In this section, we will investigate how to directly influence the power flows between the controller and system(s).

By *energy routing* we refer to all control methods that do not influence the energy content of the system. That is, the total Hamiltonian of the interconnected system is constant; the control algorithms only *direct* the energy flow. The energy flows are determined by the interconnection of the systems—e.g. of the controller and robot—so energy routing takes place in the interconnection structure: by *modulating* the Dirac structure or the internal interconnection J, whilst keeping their skew-symmetry intact.

In this section, we will give various examples of energy routing. For further applications, we refer to existing work in Duindam and Stramigioli (2004); Sanchez-Squella et al. (2010); Franken et al. (2009); Duindam et al. (2009).

3.3.1 Modulated buffer

First, we introduce the concept of a modulated buffer, which is a passive controller structure that can be regarded as a force actuator with limited energy budget. The structure of the modulated buffer is drawn in Figure 3.5: the input-state-output g of the Port-Hamiltonian System is modulated, which is equivalent to a modulated transformer element connected to a buffer, also shown in the figure. In the following theorem, the system is taken to be one-dimensional (u, y and q all in $\mathbb{R}^1$). However, the proof can be easily extended to an n-dimensional force actuator.

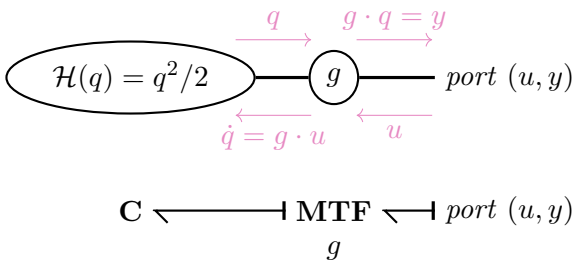


Figure 3.5: Buffer with modulated input-state-output map (top); in bond graph expression (bottom) a **C**-type storage with modulated transformer **MTF**.

Note that the *output* y of this modulated buffer will form the *input* to the system it is connected to; the buffer's *input* u is the *system's output*.

Theorem 3.1. Consider a Port-Hamiltonian System with Hamiltonian $\mathcal{H}(q) = q^2/2$ and scalar state-port map g as drawn in Figure 3.5. g is a pure modulation input and can be modulated such that the output of this PHS is an arbitrary force. The Port-Hamiltonian System is passive and the energy it can inject is bounded by $\mathcal{H}(q_0)$.

Proof. The PHS-formulation of this controller system is given by

$$\begin{aligned} \mathcal{H}(q) &= \frac{q^2}{2} \\ \dot{q} &= g \cdot u \\ y &= g \cdot q. \end{aligned} \tag{3.14}$$

With modulation

$$g = \frac{F}{q} \tag{3.15}$$

the output y of the controller is

$$y = g \cdot q = F, \tag{3.16}$$

an arbitrary force, as long as $q \neq 0$. The energy change due to g is $\partial \mathcal{H}/\partial g = 0$, so g is a purely modulating input.

Any energy injected into a system that is interconnected to the controller by

$$u_{\text{system}} = y; \quad u = -y_{\text{system}} \tag{3.17}$$

is extracted from the Hamiltonian of the controller:

$$P_{\text{system}} = u_{\text{system}} \cdot y_{\text{system}} = -yu = -\frac{\mathrm{d}\mathcal{H}}{\mathrm{d}t}. \tag{3.18}$$

When $q = 0$, the Hamiltonian becomes 0 too and the controller can no longer deliver a force that would decrease $\mathcal{H}$, i.e. the controller can no longer inject energy into the interconnected system. Therefore, the maximum energy this PHS-controller can inject is given by $\mathcal{H}(q_0)$, the initial energy in the controller. □

This modulated-buffer approach may seem like a simple but useless trick: any control output is possible, as long as there is enough energy inside the buffer. If the buffer is empty, the controller stops working. However, this approach allows to turn any control algorithm into a passive controller, giving important dissipativity and stability guarantees. Similar to the energy-tank approach, the integral dissipation inequality of (2.1) is satisfied with $S(x_0)$ equal to $\mathcal{H}(q_0)$ and $w = y^\top u$. Any nonlinear feedback problem can be easily formulated with this approach, where the result is guaranteed to be passive. In Stramigioli and Dijk (2008), this method is used to create globally attractive, energy conservative limit cycle oscillations.

3.3.2 Energy level control

Considering the importance of energy in physical systems, sometimes it may be desirable to control the actual energy level in a robotic system,

rather than a velocity, force or position. Consider for example a system that must execute a periodic motion along its natural dynamics, e.g. an oscillation: the energy level directly determines the amplitude and velocity of the system. Control of the energy present in the system is a robust and coordinate-independent way of keeping the motion on this specific (geodesic) path, with a certain desired amplitude.

Example 3.2. Consider the simple mass-spring system of Figure 3.6. A controller interacts with the system through a force/velocity port and its goal will be to maintain a certain amplitude of oscillation x_s.

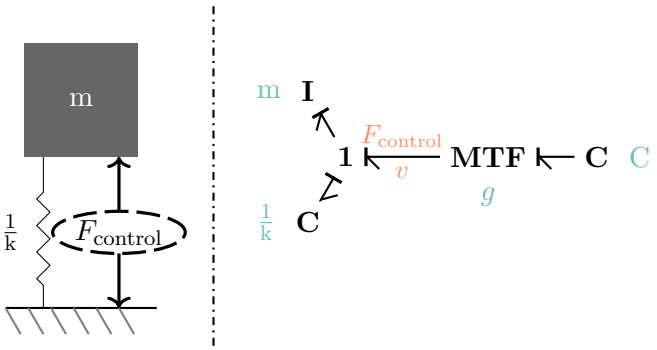


Figure 3.6: Mass-spring system with force controller, aiming to keep the energy level constant. Left: Ideal Physical Model representation; right: bond graph representation.

The PHS equations for the system are familiar:

$$\begin{aligned} \mathcal{H}(x) &= \frac{p^2}{2\mathrm{m}} + \frac{\mathrm{k}}{2} x_\mathrm{m}^2 \\ \begin{pmatrix} \dot{p} \\ \dot{x}_\mathrm{m} \end{pmatrix} &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\partial \mathcal{H}}{\partial x}(x) + \begin{pmatrix} 1 \\ 0 \end{pmatrix} F_\mathrm{control} \\ v &= \begin{pmatrix} 1 & 0 \end{pmatrix} \frac{\partial \mathcal{H}}{\partial x}(x), \end{aligned} \tag{3.19}$$

where the state vector x should not be confused with x_{m}, the mass position. The suggestion for a passive controller structure is the following:

$$\begin{aligned} \mathcal{H}_{\mathrm{c}}(q) &= \frac{q^2}{2\mathrm{C}} \\ \dot{q} &= g \cdot u \\ y &= g \cdot \frac{q}{\mathrm{C}}, \end{aligned} \tag{3.20}$$

which is the modulated storage element of Theorem 3.1, but now with a capacity C. In Figure 3.6, the controller is shown as a *bond graph* representation of the storage element C and a modulating transformer as the port-state map g. (While the bond graph formulation is not necessary for Port-Hamiltonian System theory, this example shows how well the two relate, since bond graphs are an explicit graphical representation of the various storage elements, ports and Dirac structures. We shall address bond graphs in this work no further, but refer to Paynter (1960) and Breedveld (2008).) With the interconnection $u = -v, F_{\mathrm{control}} = y$ the power injected by the controller, P_{c}, can be calculated (3.21). However, v is known—it is the input to the controller—and hence P_{c} can be *controlled*, by choosing g as follows:

$$P_{\mathrm{c}} = yv = \left(\frac{gq}{\mathrm{C}}\right) v \quad \Rightarrow \quad g := \left(\frac{\mathrm{C}}{qv}\right) P_{\mathrm{c}}. \tag{3.21}$$

Finally, by choosing P_{c} depending on the "energy error" $\tilde{E}$, the controller will inject or extract energy as needed:

$$P_{\mathrm{c}} = -\frac{\lambda}{2}\tilde{E}; \quad \tilde{E} = \mathcal{H}(x) - \frac{\mathrm{k}}{2}x_{\mathrm{s}}^2. \tag{3.22}$$

$\mathcal{H}(x)$ can be calculated from the measured v; λ is a free control parameter that determines the rate of convergence.

This control law regulates the energy level of the system with exponential convergence, resulting in the desired oscillation amplitude of x_{s}. The stability and convergence proof is given in Appendix A.

⟨*example end*⟩

3.3.3 Irreversible transduction

The dissipation matrix R in Port-Hamiltonian Systems (1.3) is usually associated with energy loss. From a mathematical point of view, this is

correct: the Hamiltonian decreases with the energy dissipated through R. Physically, however, energy is not lost; instead, dissipation means (irreversible) transduction to an entropy flow: it is converted to heat. A model for this irreversible transduction is a resistive element with two ports, the "RS" model: one mechanical with port variables (F, v) and one thermodynamical with port variables $(T, \mathrm{d}S)$, where T is the absolute temperature and $\mathrm{d}S$ the entropy flow. The constitutive relations are as follows:

$$F = -\mathrm{R}v; \quad \mathrm{d}S = \frac{\mathrm{R}v^2}{T}. \tag{3.23}$$

It is easy to check that the mechanical power $Fv = -\mathrm{R}v^2$ is always negative and equal to the power flowing out of the thermodynamical port.

The relevance for energy-based control is that this is a power-continuous element that allows energy to flow only in one direction. The ports do not necessarily have to be mechanical and thermodynamical: the port-behaviour can be used inside a controller to direct (kinetic) energy from one element to another.

Example 3.3. Consider two masses interconnected by a controller that behaves as an RS-like irreversible transduction element. Because the masses require a force input, the "thermodynamic" port relation of (3.23) has to be rearranged to output a force:

$$F_1 = -\mathrm{R}v_1; \quad F_2 = \frac{\mathrm{R}v_1^2}{v_2}, \tag{3.24}$$

which is still a power-continuous interconnection. The closed-loop equations are:

$$\mathcal{H}(x) = \frac{p_1^2}{2\mathrm{m}_1} + \frac{p_2^2}{2\mathrm{m}_2} \tag{3.25}$$

$$\dot{x} = \begin{pmatrix} 0 & -\mathrm{R}\frac{v_1}{v_2} \\ \mathrm{R}\frac{v_1}{v_2} & 0 \end{pmatrix} \frac{\partial \mathcal{H}}{\partial x}(x), \tag{3.26}$$

with the two masses' momenta as the state vector. From (3.24), we would usually find "$-\mathrm{R}$" in the upper-left position of the matrix in (3.26), but the upper-right term gives the same result:

$$-\mathrm{R}\frac{v_1}{v_2}\frac{\partial \mathcal{H}}{\partial p_2} = -\mathrm{R}\frac{v_1}{v_2}\frac{p_2}{\mathrm{m}_2} = -\mathrm{R}\frac{v_1}{v_2}v_2 = -\mathrm{R}v_1. \tag{3.27}$$

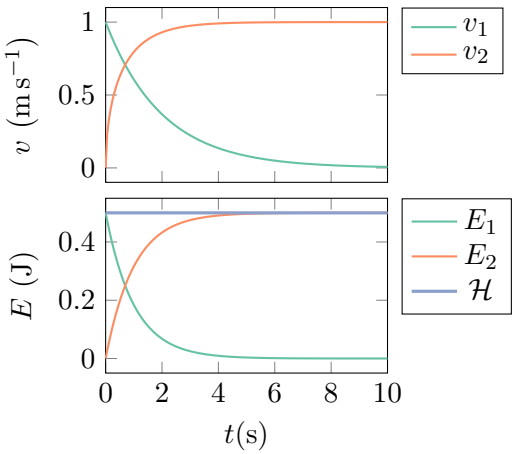


Figure 3.7: Simulation of two masses interconnected by a power-continuous controller that creates irreversible transduction from mass 1 to mass 2. For mass 1, the controller behaves as a friction, but all "dissipated" energy is diverted to mass 2. (E_i is the kinetic energy of mass i.)

Now, however, the matrix is skew-symmetric and thus the system is lossless—because the damping energy is not dissipated, but *directed* to the other mass.

Figure 3.7 shows a simulation of this example, with $\mathrm{m}_1 = \mathrm{m}_2 = 1\,\mathrm{kg}$, $\mathrm{R} = 0.5\,\mathrm{N\,s\,m^{-1}}$ and $x(0) = (1 \;\; 0)^\top \mathrm{kg\,m\,s^{-1}}$. Mass 1 is slowed down by the friction of R, but the dissipated energy is injected into mass 2. The total energy remains constant. ⟨*example end*⟩

3.3.4 Kinetic energy routing

The concept of irreversible transduction can be generalised further, to achieve arbitrary energy routing in the controller. A gyrator is a power-continuous Dirac structure that connects two Port-Hamiltonian Systems as follows:

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{bmatrix} 0 & r \\ -r & 0 \end{bmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}, \tag{3.28}$$

with r the gyration ratio. It is easy to verify that the connection is power-continuous, but more importantly, the power flow through the interconnection is modulated by r.

Theorem 3.2. Given two Port-Hamiltonian Systems Σ_1 and Σ_2 with power ports (u_1, y_1) and (u_2, y_2) that are interconnected by a gyrator with gyration ratio r. If the gyrator is modulated according to

$$r = \frac{1}{y_2 y_1} \cdot r_P, \tag{3.29}$$

then the energy flow from Σ_2 to Σ_1 is equal to r_P, as long as $y_1 \neq 0$ and $y_2 \neq 0$. This interconnection is power-continuous and r_P is a pure modulation to passively direct the energy flow between the two systems.

Proof. With the gyrating Dirac structure behaving according to (3.28), the power flows into Σ_1 and Σ_2, P_1 and P_2, are:

$$P_1 = y_1 u_1 = y_1 r y_2; \quad P_2 = y_2 u_2 = -y_2 r y_1. \tag{3.30}$$

$P_1 = -P_2$, so the interconnection is power-continuous. With the gyration ratio given by (3.29), P_1 is:

$$P_1 = y_1 u_1 = y_1 r y_2 = y_1 \frac{1}{y_2 y_1} r_P y_2 = r_P. \tag{3.31}$$

□

Corollary 3.3. If the gyration ratio is instead set according to (3.32), the ratio is always well-defined—even if $y_1 = 0$ or $y_2 = 0$. However, for $y_1 = 0$ or $y_2 = 0$, still no energy transfer takes place; and while the direction of power flow is still controlled by r_P, the rate of energy exchange depends on y_1 and y_2 (3.33).

$$r = y_1 y_2 r_P \tag{3.32}$$

$$P_1 = y_1 u_1 = y_1 r y_2 = y_1 (y_1 y_2) r_P y_2 = (y_1^2 y_2^2) r_P. \tag{3.33}$$

In practice, the condition that $y_1 \neq 0$ and $y_2 \neq 0$ is rather restrictive and the alternative gyration ratio does not allow output-independent control over the rate of energy exchange. However, as is shown in the following example, there is a way to get around the restriction, with little deviation between r_P and the actual energy flow.

Example 3.4. Consider a simple example of two masses connected to a control system that implements the energy routing of Theorem 3.2, illustrated in Figure 3.8. The inputs u_i represent a force; the corresponding outputs y_i the masses' velocities.

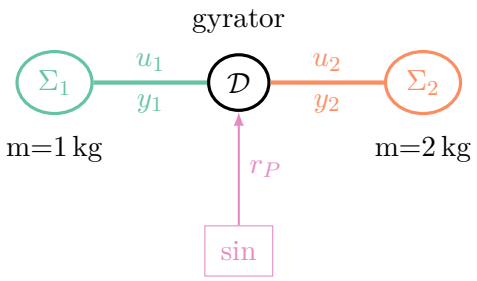


Figure 3.8: Two masses connected by a gyrating controller that can route kinetic energy. The "`sin`" block generates a sinusoidal energy flow setpoint.

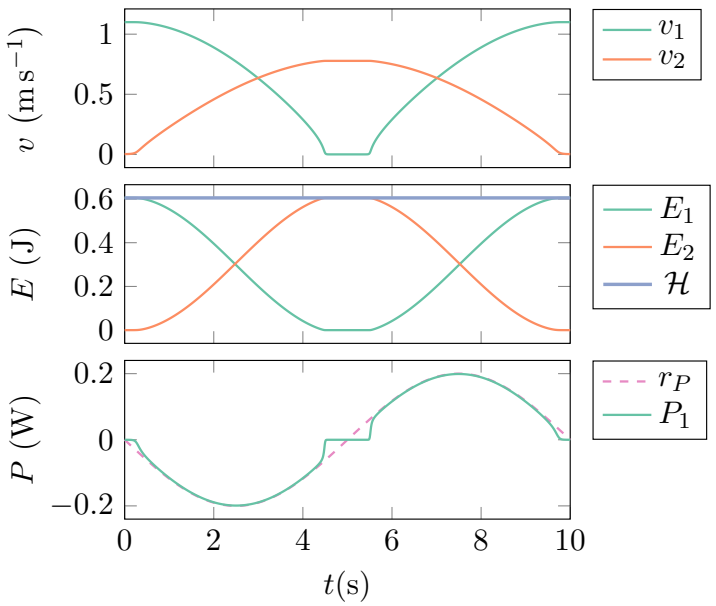


Figure 3.9: Implementation of the kinetic energy router of example Example 3.4. Because of the adapted gyration ratio (3.34), when v_1 or v_2 gets close to 0, the actual power flow P_1 deviates from r_P.

In order to allow situations where $y_1 = v_1 = 0$ and $y_2 = v_2 = 0$, the gyrator ratio of (3.29) is adapted to:

$$r = \frac{y_1 y_2}{(y_2^2 + \epsilon)(y_1^2 + \epsilon)} r_P \tag{3.34}$$

where $\epsilon > 0$ is some small number. For large values of y_1 and y_2—that is, $y_i^2 \gg \epsilon$—the gyration ratio will be approximately equal to (3.29). When $y_i = 0$, it remains well-defined.

The system of Figure 3.8 is simulated with an initial velocity of $v_1 = 1.1\,\mathrm{m\,s^{-1}}$ and r_P is a sine wave of amplitude 0.2 W and period 10 s. ϵ is set to 0.001. Figure 3.9 shows the results: except for when $v_1 \approx 0$ or $v_2 \approx 0$, the actual energy flow to Σ_1 is equal to the setpoint r_P. The energy plot in the middle shows that the complete system is power-continuous, as was expected. ⟨*example end*⟩

The previous example considered two simple systems Σ_1 and Σ_2. However, the concept of energy routing can be applied to much more complex systems. For example, the two ports do not have to be the velocity of two separate masses. It can also be the two velocity components of a mass moving in 2-D, or even those velocities in a transformed coordinate system, such that v_1 corresponds to the mass's velocity in a desired direction and v_2 the velocity perpendicular to that. By modulating r_P, all kinetic energy[2] can be routed from the "unwanted" direction to the desired direction, thereby steering the mass along a certain trajectory—without using external energy. In Duindam and Stramigioli (2004), this method is used to create passive limit cycles.

In Folkertsma et al. (2014), the gyrating controller is connected to the stiffness-changing ports of two springs $(\dot{K}, x^2/2)$, adapting their stiffness in a power-continuous way to achieve synchronisation between two oscillators by routing energy between the springs.

[2] Full energy transfer does require the systems, or at least the "source" system, to be zero-state-detectable.

4

Control by *physical* interconnection

In control by interconnection, the control algorithm has the dynamics of a Port-Hamiltonian System that could represent a physical system, e.g. the virtual springs and masses of Example 3.1. Similarly, in Li and Horowitz (1999), controllers are designed as passive systems with a physical interpretation of a flywheel or a spring: a direct virtual analogue of a physical element. At the same time, the physical system can be represented as a PHS. The interconnection of the controller and system is again a Port-Hamiltonian System, which raises the question: what distinguishes the controller from the physical system? The answer is: conceptually and mathematically, nothing at all.

This means that the boundary between system and controller is a fluid, shifting boundary that is frozen only at the moment of implementation, when the interface between computer and physical system is decided. And even then, one might argue that part of the control effort is carried out by the system dynamics, or part of the system dynamics are simulated in the controller. In the previous section, we have studied control by interconnection, where the controller was represented by a Port-Hamiltonian System, which gives a physical interpretation to the control algorithms. In this section, we shall consider the physical

system's PHS-representation in the light of the control objectives. We call this "Control by *physical* interconnection," where desired behaviour previously achieved by the controller is now achieved by the physical system itself.

4.1 Physical compliance

Electric motors have many benefits when used in robotic systems: high availability in a broad range, easy electrical interconnection to the control system, integrated position encoders, a large selection of gearboxes, precise torque control, easy integration in rotational joints, et cetera. However, there is also a major drawback: electric motors are very efficient in a high speed/ low torque region, whereas many robotic systems usually operate in low speed/high torque regions.

An electric motor's efficiency is mostly determined by its motor constant K_m and its electrical resistance R_e, for a given required torque output. In low speed regions, mechanical losses are negligible and the electrical loss is computed as:

$$P_{\mathrm{e,loss}} = i^2 R_\mathrm{e} = \left(\frac{\tau}{K_\mathrm{m}}\right)^2 R_\mathrm{e} = \frac{\tau^2}{S}. \tag{4.1}$$

S is the "motor steepness" and defined as $K_\mathrm{m}^2/R_\mathrm{e}$. Clearly, a higher motor steepness results in lower power loss. In motor fabrication, there is a tradeoff between size, weight and motor steepness: to increase S, the resistance must be low (thick wires), or the motor constant high (long wires, large motor diameter, high magnetic fields). For research on the analysis and improvement of electric motor efficiency, we refer to Seok et al. (2015) and Dresscher (2016).

In practice, the motor with lowest losses may be picked, but there is still no way around the fact that for zero speed and nonzero torque—which is the desired equilibrium in e.g. position control—the motor efficiency becomes zero:

$$\eta_\mathrm{motor} = \frac{P_\mathrm{mechanical}}{P_\mathrm{loss}} = \frac{\omega\tau}{P_\mathrm{e}} \underset{\omega=0,\tau\neq 0}{=} \frac{0}{\tau^2/S}. \tag{4.2}$$

4.1.1 Physical storage elements

Much of the work that electric motors do in robotic applications is conservative, for example when moving a robot arm up and down in a gravity field. Indeed, if for stability or safety reasons the controller is strictly passive, as argued in §2.1.1, the net mechanical work of the motor is 0. Therefore, much if not all of the work that the motor does could be done by a passive physical storage element: by a spring or by an inertia. (Indeed, in Li and Horowitz (1999), controllers are exactly a virtual spring or flywheel.)

A flywheel stores kinetic energy, determined by its state, the momentum p:

$$E_{\text{flywheel}}(p) = \frac{p^2}{2J}, \tag{4.3}$$

which is in value equal to the kinetic co-energy E^*:[1]

$$E^*_{\text{flywheel}}(\omega) = \frac{J\omega^2}{2}. \tag{4.4}$$

J is the flywheel's moment of inertia. While flywheels are used in automotive industry (Lukic et al. (2008)), for example to make stops and starts of buses more efficient, they are not so suitable for robotics applications. This is because they store and release their energy typically at high speeds, which is exactly the region in which electric motors are quite efficient, and in which robotic systems typically do not operate.[2]

Springs, on the other hand, store potential energy, determined by the state q:

$$E_{\text{spring}}(q) = \frac{Kq^2}{2} \tag{4.5}$$

or potential co-energy

$$E^*_{\text{spring}}(F) = \frac{F^2}{2K}, \tag{4.6}$$

where K is the spring stiffness. Because the spring can store and release energy at low or even zero velocity, it is very suitable for application in

[1] The co-energy is a function of ω, the port variable that is a thermo-dynamical intensive variable, rather than a function of (the extensive variable) p.

[2] As electrical energy storage, they are very suitable: through the gyrating motor/dynamo, the flywheel behaves as a capacitor-like storage element as seen from the electrical side.

robotics, where it can assist electric motors in regions where the motor efficiency is very low. Furthermore, this energy is stored in a physical deformation in the mechanical domain, at zero velocity. Therefore, the energy is not lost to friction over time, as may be the case in a flywheel.

Equations (4.3) and (4.5) give the internal energy of the storage elements, i.e. the Hamiltonian. The elements may be interconnected to a robotic system through a fixed or variable gearbox—the latter will modulate their output in a similar way to the *virtual* modulated buffer of §3.3.1. A spring with a modulated output is one way of achieving variable stiffness; see §4.2. Also with a fixed interconnection, the addition of springs can have energetic benefits, as shown in the next two sections.

4.1.2 Parallel spring configuration

When the spring is connected in parallel to the motor, their torques are added:

$$\tau_{\text{tot}} = \tau_{\text{motor}} + \tau_{\text{spring}} = \tau_{\text{motor}} + K(q - q_0). \tag{4.7}$$

The spring is characterised by two parameters: the spring stiffness K and its rest length q_0. The spring can be dimensioned to supply quasi-static torque, for example for passive gravity compensation. In that case, a low K combined with high q_0 leads to a nearly constant output torque, greatly lowering the static torque on the motor and thereby reducing electrical energy loss. In more dynamic applications, especially when the motion executed by the robot is periodic, an optimal spring can even deliver a large part of the required torque during the trajectory. In Hunt et al. (2016) and Wanders et al. (2015), parallel stiffness is applied in energy-efficient hopping robots. The use of balancing springs has been also extensively and elegantly treated in Herder (1998).

Example 4.1. Consider the simple 2-DoF robot arm in Figure 4.1, consisting of two rotational joints with joint angle q_i and joint torque τ_i. Its end-effector follows an elliptical trajectory according to (4.8) in the (x, y)-plane, so there is no gravity interaction. The system parameters can be found in Table 4.1.

$$p_{\text{setpoint}} = \begin{pmatrix} 0.5 + 0.8\cos(\pi t) \\ 1.2 + 0.3\sin(\pi t) \end{pmatrix} \text{m}. \tag{4.8}$$

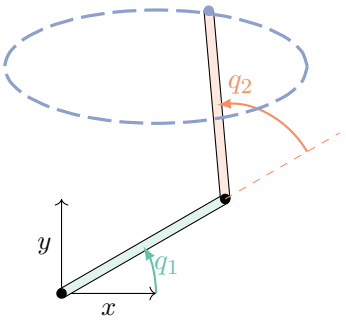


Figure 4.1: A simple pick-and-place 2-DoF robot arm whose end-effector moves along an elliptical trajectory.

Parameter	Value		Description
L	1	m	Link length
m	1	kg	Link mass
K_m	0.04	$\mathrm{N\,m\,A^{-1}}$	Motor constant
R_e	1	Ω	Motor electrical resistance
n	1:33		Gear reduction ratio
K	30	$\mathrm{N\,m^{-1}}$	Impedance controller stiffness
D	3	$\mathrm{N\,s\,m^{-1}}$	Impedance controller damping
k_1	15	$\mathrm{N\,m\,rad^{-1}}$	Optimal stiffness joint 1
$q_{0,1}$	0.5	rad	Optimal rest length joint 1
k_2	2.6	$\mathrm{N\,m\,rad^{-1}}$	Optimal stiffness joint 2
$q_{0,2}$	1.95	rad	Optimal rest length joint 2

Table 4.1: Parameters of the 2-DoF pick-and-place robot arm with parallel springs.

The controller used in this example is a virtual prismatic spring and damper between the end-effector and the setpoint, with parameters according to Table 4.1:

$$\begin{pmatrix}\tau_1\\ \tau_2\end{pmatrix} = J^\top(q)\cdot\left(K(p_\mathrm{setpoint}-p_\mathrm{ee})+D(\dot{p}_\mathrm{setpoint}-\dot{p}_\mathrm{ee})\right). \tag{4.9}$$

$J(q)$ is the Jacobian map from joint velocity $\dot{q}$ to end-effector velocity $\dot{p}_{\text{ee}}$, and so $J^\top(q)$ is the dual map from end-effector force to joint torque. The performance of this controller will be rather poor, with no feed-forward torque for acceleration, but the end-effector does follow the trajectory well enough to show the effect of parallel springs.

A simple linear spring is added to both joints, according to (4.7). The motor torque is set to

$$\tau_{\text{motor},i} = \tau_i - K_i(q_i - q_{0,i}) \tag{4.10}$$

such that the total torque applied on the joint is always τ_i.

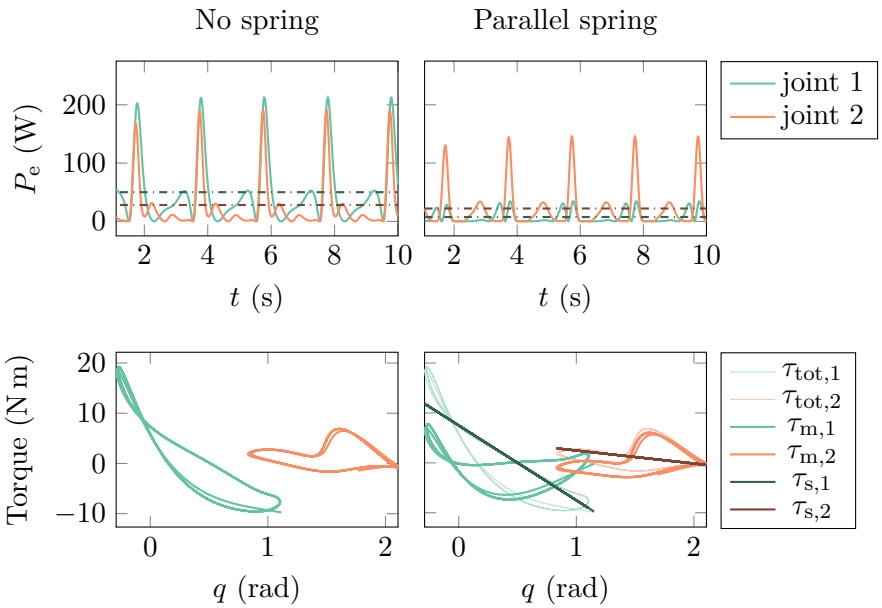


Figure 4.2: Simulation of the pick-and-place robot arm that executes a periodic trajectory. The power plotted is the electrical power loss $i^2 R_e$; dashed in dark is the average power loss. Adding a parallel spring greatly reduces this loss, especially for the first joint. The motor torque is lowered by the spring torque, seen in the bottom-right.

With a spring stiffness of $k_i = 0$, i.e. without a parallel spring, both joints' motors show peaks in power loss of up to 200 W (Figure 4.2). The average power loss for the two motors is 50 W and 28 W, respectively.

From the joint-space plots of both motors, it can already be seen that especially motor 1 delivers a torque that is largely proportional to its rotation, which means that a linear parallel spring could supply most of the work. An numerical optimisation on parameters k_i and $q_{0,i}$ that minimises the objective function (4.11) gives two optimal parallel springs as listed in Table 4.1.

$$J = \int_0^T (P_{\mathrm{e},1} + P_{\mathrm{e},2}) \, \mathrm{d}t. \tag{4.11}$$

From the plots on the right hand side of Figure 4.2 it is clear that the springs significantly reduce the energy loss: for motor 2 from 28 W to 22 W, a moderate improvement, but for motor 1 from 50 W to 7 W. Overall, this means the energy loss is reduced by 63 %. Moreover, the maximum torque delivered by the motors has been lowered, which allows for lighter and smaller motors to be used.

In Plooij and Wisse (2012), the use of nonlinear parallel springs is investigated on a realistic pick-and-place 2-DoF robot arm, where real-world experiments show an efficiency improvement of 20 %. ⟨*example end*⟩

In Folkertsma et al. (2012) it was shown that the inclusion of parallel springs in the MITCheetah quadruped robot would lead to a reduction in power consumption by over 50 %. Hunt et al. (2016) have obtained very efficient resonance-based kangaroo-like walking robots by applying parallel springs.

4.1.3 Serial spring configuration

One way to make the motor operate in a more efficient region is by including a large gear reduction. While this solves the problem of low speed, high torque applications, it comes with two major drawbacks: first, gearboxes with high ratios often have a rather low efficiency; second, the reflected motor inertia is scaled by n^2, which may lead to inefficient power transfer (through bad inertia matching), low bandwidth and bad interaction behaviour. Indeed, the motor can become practically nonbackdriveable, which makes impedance or even torque

control infeasible. Even if the motor is backdriveable, the friction behaviour of a gearbox is typically nonlinear, which makes precise torque control difficult.

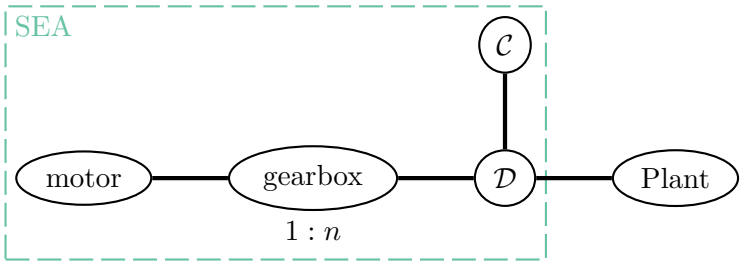


Figure 4.3: A Series Elastic Actuator allows the motor to operate in a more efficient region, but more importantly, it allows for precise force control and physical, reversible energy storage. The Dirac structure is such that $\tau_{\text{gearbox}} = \tau_{\text{out}} = \tau_{\text{spring}}$ and consequently $\omega_{\text{spring}} = \omega_{\text{gearbox}} - \omega_{\text{out}}$.

The solution to most of these problems is to connect a spring in series with the motor, resulting in a Series Elastic Actuator, as introduced by Pratt and Williamson (1995). A Port-Hamiltonian representation is show in Figure 4.3. Because the output torque is determined purely by the spring state and stiffness, according to (4.12), force control is transformed into an (easier) position control problem.

$$\tau_{\text{out}} = K \cdot (q_{\text{out}} - q_{\text{motor}}/n). \tag{4.12}$$

If the gear reduction ratio is sufficiently high, such that the motor is practically nonbackdriveable, any static holding torque—that is to say, $\dot{\tau}_{\text{out}} = 0$ and $\dot{q}_{\text{out}} = 0$—can be sustained with zero power draw by the motor.

Another benefit of SEA is that the (large, reflected) motor inertia is decoupled from the output. Therefore any impacts on the robot, for example ground contact events in a walking robot, are absorbed by the spring. On the one hand this makes the robot safer and lowers energy loss; on the other hand there is less strain on the gearbox and motor.

However, the decoupling between motor and output also mean that the achievable bandwidth is much lower. This is most easily seen by considering the classical transfer function from motor torque τ_{M} to load

angle θ_{L} for the simplified system of Figure 4.4:

$$J_{\mathrm{L}} s^2 \Theta_{\mathrm{L}} = K(\Theta_{\mathrm{M}} - \Theta_{\mathrm{L}}) \tag{4.13}$$

$$J_{\mathrm{M}} s^2 \Theta_{\mathrm{M}} = -K(\Theta_{\mathrm{M}} - \Theta_{\mathrm{L}}) + T_{\mathrm{M}} \tag{4.14}$$

$$\Theta_{\mathrm{L}} = \frac{K}{(J_{\mathrm{M}} s^2 + K)(J_{\mathrm{L}} s^2 + K) - K^2} T_{\mathrm{M}} \tag{4.15}$$

$$= \frac{1}{(J_{\mathrm{L}} J_{\mathrm{M}} s^2 / K + (J_{\mathrm{M}} + J_{\mathrm{L}}))\, s^2} T_{\mathrm{M}}. \tag{4.16}$$

For $K \to \infty$, the transfer function reduces to $\frac{1}{(J_{\mathrm{M}}+J_{\mathrm{L}})s^2}$, i.e. a direct-drive. With a low K, however—which is whence the SEA derives its benefits—extra low-pass behaviour is introduced by the term $J_{\mathrm{L}} J_{\mathrm{M}} s^2 / K$.

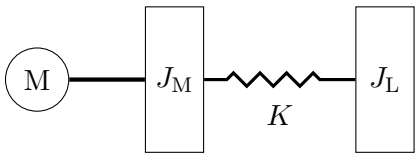


Figure 4.4: A simplified SEA model with a spring between motor and load inertia. The bandwidth of this system is—depending on K—much lower than a direct-drive actuation.

4.2 Variable stiffness

From the previous section it is clear that adding physical storage elements, especially in the form of springs, can be very beneficial for energy-efficiency, precise torque control, impact absorption, et cetera. However, the optimal stiffness very much depends on the application.

In the case of parallel elastic actuation, a spring that is used to compensate gravity in a robot arm only counteracts the robot's weight in a specific configuration, or at best in a limited range of arm positions. In series elastic actuation, there is a trade-off between desiring a low spring stiffness for more precise force control and better impact absorption, versus a higher stiffness for more precise position control and higher bandwidth.

In highly dynamic applications, such as running legged robots, or robot arms executing fast, periodic motions, the spring stiffness should be tuned to the robot inertia and the desired period of the motion.

It follows that the spring stiffness can be optimised for a certain situation, trajectory, or application; but once the robot deviates from this nominal behaviour, a different stiffness would be optimal. With a *variable* stiffness, the stiffness can be tuned so it is optimal in every situation (Visser et al. (2011b)). Therefore, the concept of actuators with an adaptable stiffness, called *Variable Stiffness Actuators* or VSAs, has received much attention recently. We refer to Ham et al. (2009); Vanderborght et al. (2009, 2013); Wolf et al. (2016) for extensive reviews of the state-of-the-art in VSAs.

A VSA has a physical spring and a means to change the stiffness felt at the output of the actuator. To distinguish VSAs from Variable Spring Mechanisms, they are also understood to have an actuator capable of injecting energy into the system. The output port of the VSA is given by the power pair $(\tau_{\text{out}}, \dot{r})$, with τ_{out} the torque at the output and r the output displacement. Because the spring is most often in a serial configuration, the VSA's output is the torque. The apparent output stiffness is a local property defined as

$$k_{\text{out}} = \frac{\delta \tau_{\text{out}}}{\delta r}. \tag{4.17}$$

Example 4.2. In its simplest form, a VSA can be represented by a linear spring with mechanical control port "p1", a stiffness-changing control port "p2" and an interaction or output port $(\tau_{\text{out}}, \dot{r})$. The Port-Hamiltonian System of this VSA is given by

$$\begin{aligned} \mathcal{H}(x) &= \frac{k}{2} s^2 \qquad (4.18) \\ \dot{x} = \begin{pmatrix} \dot{s} \\ \dot{k} \end{pmatrix} &= (0) \frac{\partial \mathcal{H}}{\partial x}(x) + \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{r} \end{pmatrix} \\ y = \begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_{\text{out}} \end{pmatrix} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} ks \\ s^2/2 \end{pmatrix}, \end{aligned}$$

where the state $x = (s, k)^\top$ consists of both the spring's strain s and its stiffness k. The interaction ports are $(\tau_1, \dot{q}_1)$, which may inject mechanical power into the spring; $(\tau_2, \dot{q}_2)$, the stiffness-changing port; and $(\tau_{\text{out}}, \dot{r})$, the output port.

Note that the input-state-output map g need not be constant as in this example; nor have the two control ports to be decoupled—the choice is made in this example such that port 1 only injects mechanical energy, through the direct coupling from $\dot{q}_1$ to $\dot{s}$; and port 2 only changes the stiffness, through $\dot{k} = \dot{q}_2$.

From these PHS equations it is apparent that changing the stiffness is not "free": there is an energy cost from the power injected at port 2,

$$P_2 = \tau_2 \dot{q}_2 = g_2^\top \frac{\partial \mathcal{H}}{\partial x}(x)\dot{q}_2 = \frac{s^2}{2}\dot{k}. \tag{4.19}$$

Only when the spring is unloaded there is no energy cost associated with changing k. (This can be a drawback, but also be exploited; see Folkertsma et al. (2014).)

It is of course possible to control $\dot{q}_1$ and $\dot{q}_2$ in such a way that the internal energy of the VSA is not changed, as long as the output does not move ($\dot{r} = 0$):

$$P_1 = -P_2 \Rightarrow ks\dot{q}_1 = -\frac{s^2}{2}\dot{q}_2 \Rightarrow \dot{q}_1 = -\frac{s}{2}\dot{q}_2. \tag{4.20}$$

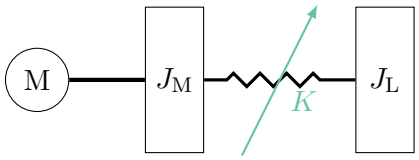
If a real VSA can be constructed or controlled in such a way that the energy cost associated with changing the stiffness is *always* 0, we call that VSA *energy-efficient*. ⟨*example end*⟩

A VSA generally has two motors, because the output force of the spring has two degrees of freedom, r_0 and k. The two motors can be configured in two ways (Wolf et al. (2016)):

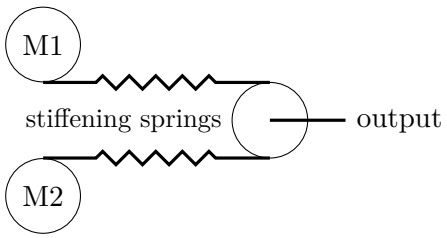
1. an antagonistic motor setup, as in Figure 4.5b;
2. an independent motor setup, as in Figure 4.5c.

The stiffness can be varied by means of three methods:

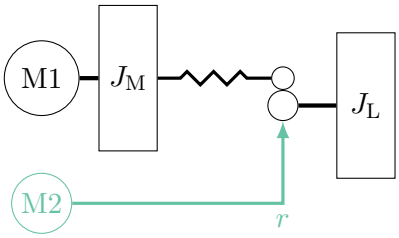
1. variation of the spring preload, as in Figure 4.5b;



(a) The actual physical stiffness of the compliant element may be changed.



(b) An antagonistic motor setup with stiffening (non-linear) springs results in an output stiffness depending on the preload.



(c) A mechanism, usually a variable gear ratio between spring and output, can change the apparent output stiffness of the spring.

Figure 4.5: Three ways of achieving a variable-stiffness series elastic element in a Variable Stiffness Actuator (VSA).

2. variation of the transmission ratio between spring and output (Figure 4.5c);

3. influence of the spring's physical properties (Figure 4.5a).

These stiffness-changing methods can all be combined with either of the motor configurations; Figure 4.5 shows three of the six possible combinations. We will focus on the configurations of Figure 4.5c and 4.5b, because they are used often and are interesting from an energy perspective.

4.2.1 Antagonistic stiffening springs

The VSA configuration of Figure 4.5b features two antagonistic motors, connected to the output pulley by nonlinear, stiffening springs. The motor positions are denoted by q_i, output position by r and spring states by s_i. In spite of what the drawing suggests, both motors are

connected such that a positive q_i means a compression of the spring. The power-conjugate torques of each port are τ_i for the motors and τ_{out} for the output link.

It is known that to obtain a linear output stiffness, the spring force must be quadratic:

$$\tau_i = -k_i s_i |s_i| = -k_i s_i^2, \tag{4.21}$$

where the simplification can be made because in an antagonistic spring set-up the springs are always loaded in extension, i.e., $s_i > 0$.

The apparent output stiffness and the energy storage of the actuator can be studied in the Port-Hamiltonian formulation of this system. The state vector is $x = (s_1, s_2)^\top$ and the system has three power ports: $(\tau_1, \dot{q}_1)$, $(\tau_2, \dot{q}_2)$ and $(\tau_{\text{out}}, \dot{r})$.

$$\begin{aligned}
\mathcal{H}(x) &= \frac{k}{3}\left((s_1)^3 + (s_2)^3\right) \\
\begin{pmatrix}\dot{s}_1 \\ \dot{s}_2\end{pmatrix} &= (0)\frac{\partial \mathcal{H}}{\partial x}(x) + \begin{pmatrix}-1 & 0 & -1 \\ 0 & -1 & 1\end{pmatrix}\begin{pmatrix}\dot{q}_1 \\ \dot{q}_2 \\ \dot{r}\end{pmatrix} \\
\begin{pmatrix}\tau_1 \\ \tau_2 \\ \tau_{\text{out}}\end{pmatrix} &= \begin{pmatrix}-1 & 0 \\ 0 & -1 \\ -1 & 1\end{pmatrix}\begin{pmatrix}k(s_1)^2 \\ k(s_2)^2\end{pmatrix}.
\end{aligned} \tag{4.22}$$

The apparent output stiffness is given by

$$k_{\text{out}} = \frac{\partial \tau_{\text{out}}}{\partial r} = g_r^\top \frac{\partial^2 \mathcal{H}(x)}{\partial x^2} g_r, \tag{4.23}$$

where g_r denotes the third column of g, the mapping from $\dot{r}$ to $\dot{s}$. The stiffness is found to be

$$k_{\text{out}} = \begin{pmatrix}-1 & 1\end{pmatrix}\begin{pmatrix}2ks_1 & 0 \\ 0 & 2ks_2\end{pmatrix}\begin{pmatrix}-1 \\ 0\end{pmatrix} = 2k(s_1 + s_2). \tag{4.24}$$

If the system starts at rest with $q_i = r = s_i = 0$, then $s_1 = -q_1 - r$ and $s_2 = -q_2 + r$, and the stiffness expressed in motor positions is

$$k_{\text{out}} = -2k(q_1 + q_2). \tag{4.25}$$

From $s_i > 0$ follows that $(q_1 + q_2) < 0$ and thus this output stiffness is always positive and linearly dependent on the total pretension $q_1 + q_2$.

The equilibrium position r_0 of the output is given by

$$\tau_{\text{out}}|_{r=r_0} = 0 \Rightarrow k(-q_1 - r_0)^2 = k(-q_2 + r_0)^2 \Rightarrow r_0 = \frac{q_2 - q_1}{2}, \quad (4.26)$$

which is the average differential motor position: a simple linear coordinate transformation on q_1 and q_2 to their sum and difference gives direct control inputs for k_{out} and r_0.

While the derivations show that an easily controllable VSA can be realised in this way, it is also apparent that there is always a load on the motor, even if $\tau_{\text{out}} = 0$, due to the preload on the springs:

$$\tau_i = -k(s_i)^2 \underset{r=r_0}{=} k\left(-\frac{1}{2}(q_1 + q_2)\right)^2 = k\left(-\frac{1}{2}\left(-\frac{k_{\text{out}}}{2k}\right)\right)^2 = \frac{k_{\text{out}}^2}{16k}. \quad (4.27)$$

Considering the significant electrical energy loss in motors, energy-efficiency-wise this is not an ideal situation. Furthermore, not all energy stored in the springs can be used by the system: the output can extract energy until the minimum of $\mathcal{H}$ for r, i.e. until

$$\frac{\partial \mathcal{H}}{\partial r} = g_r^\top \frac{\partial \mathcal{H}}{\partial x}(x) = 0 \Rightarrow k(s_1)^2 = k(s_2)^2 \Rightarrow r = r_0. \quad (4.28)$$

However, in the equilibrium position $r = r_0$ there is still energy in the actuator, namely

$$\mathcal{H}|_{r=r_0} = \frac{2k}{3}\left(-\frac{q_1 + q_2}{2}\right)^3 = \frac{2k}{3}\left(\frac{k_{\text{out}}}{4k}\right)^3. \quad (4.29)$$

Due to strain or stress limits caused by the mechanism or physical properties of the springs, there is a maximum to the energy that can be stored in each spring. This means that the higher the output stiffness, the less energy storage is available to the system coupled to the actuator:

$$\mathcal{H}_{\text{available}} = \mathcal{H}_{\max} - \mathcal{H}_{r=r_0}. \quad (4.30)$$

4.2.2 Variable transmission

The variable-transmission VSA as shown in Figure 4.5c is promising from an energy perspective, because the output is coupled directly to the spring, as is the main motor M1. This means that all of the energy

storage capacity of the spring can be used; the system truly behaves as a series elastic actuator with variables spring stiffness. Furthermore, the stiffness changing mechanism for M2 can be designed such that it is completely decoupled from the spring, input and output, such that the stiffness can be changed using minimal—theoretically even 0—energy.

We shall analyse a 1-DoF VSA with a variable transmission by studying properties of the Dirac structure that defines the interconnection between the motors, internal states and output. In Visser et al. (2011a), the analysis is done for n-DoF VSAs and used to design an energy-efficient Variable Stiffness Actuator.

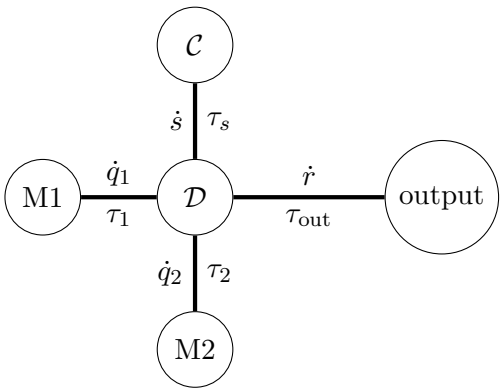


Figure 4.6: Port-Hamiltonian representation of the variable-transmission type VSA. There is only one storage element, the spring, but the Dirac structure has an internal degree of freedom, q, that represents the variable transmission between spring and output.

The drawing in Figure 4.6 defines all the port variables of the VSA. Their interconnection is determined by the Dirac structure $\mathcal{D}$, which may depend on the output position r and some internal degree of freedom of the mechanism, q that may in turn depend on q_i or s. A matrix representation of the Dirac structure is given in (4.31). Each α_i may be $\alpha_i(q, r)$. The lower-right part of the matrix is filled with zeros, because that part describes a gyrating effect between the mechanical ports, which is impossible. The matrix is skew-symmetric

so the interconnection is power-continuous.

$$\begin{pmatrix} \dot{s} \\ \tau_1 \\ \tau_2 \\ \tau_{\text{out}} \end{pmatrix} = \begin{bmatrix} 0 & \alpha_1 & \alpha_2 & \alpha_r \\ -\alpha_1 & 0 & 0 & 0 \\ -\alpha_2 & 0 & 0 & 0 \\ -\alpha_r & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \tau_s \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{r} \end{pmatrix}. \tag{4.31}$$

With this interconnection matrix, the PHS equations of the system with state $x = (s, q)^\top$ and external ports for M1, M2 and the output are:

$$\mathcal{H} = \frac{k}{2}s^2 \tag{4.32}$$

$$\begin{pmatrix} \dot{s} \\ \dot{q} \end{pmatrix} = (0)\frac{\partial \mathcal{H}}{\partial x}(x) + \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_r \\ \beta_1 & \beta_2 & \beta_r \end{pmatrix} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{r} \end{pmatrix}$$

$$\begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_{\text{out}} \end{pmatrix} = \begin{pmatrix} \alpha_1 & \beta_1 \\ \alpha_2 & \beta_2 \\ \alpha_r & \beta_r \end{pmatrix} \frac{\partial \mathcal{H}}{\partial x}(x),$$

where β_i determines the mapping of inputs to the internal configuration of the VSA.

In order to separate the functions of M1 and M2 into a pure power source and pure stiffness changing motor, respectively, $\dot{q}_2$ must be decoupled from $\dot{s}$ ($\alpha_2 = 0$) and $\dot{q}$ must only depend on $\dot{q}_2$ ($\beta_1 = \beta_r = 0$). If the mechanism can be constructed in such a way, the result is a perfectly energy-efficient VSA with the following properties.

The apparent output stiffness around a certain position $r = \bar{r}$ is found by

$$k_{\text{out}}(q, r = \bar{r}) = g_r^\top \frac{\partial^2 \mathcal{H}}{\partial x^2} g_r = (\alpha_3(q, \bar{r}))^2 k \tag{4.33}$$

and thus the output stiffness can be changed if the mechanism is such that α_3 depends on q, i.e., such that the transmission ratio between output and spring is variable; $\dot{q} = \beta_2 \dot{q}_2$. The energy cost associated with a stiffness change is

$$P_2 = \tau_2 \dot{q}_2 = g_2 \frac{\partial \mathcal{H}}{\partial x}(x)\dot{q}_2 = \begin{pmatrix} 0 & \beta_2 \end{pmatrix} \begin{pmatrix} ks \\ 0 \end{pmatrix} \dot{q}_2 = 0. \tag{4.34}$$

Finally, the equilibrium position of the output r_0 when $\tau_{\text{out}} = 0$ depends on α_1 and α_3, but is determined by M1.

4.3 Morphological computation

The concept of control by *physical* interconnection can be taken further by pushing the boundary between control and system even farther towards the physical system. The "computation" that is traditionally carried out by a computer running control algorithms is mostly—or even completely—executed in the controlled system. This sharing of control of the robot between "traditional" control and the *morphology* of the robot itself is called *morphological computation*, introduced by Pfeifer et al. (2007).

In general, morphology may refer to only the shape of the mechanical system, in which case there is only *kinematic* morphological computation. A famous example of this is the Strandbeest Jansen (2016), which contains a mechanism that converts a continuous rotating motion into a walking leg motion. This and similar mechanisms were analysed in Sitharam and Wang (2014). While a well-designed kinematic morphology results in kinematic morphological computation, it is "merely" good mechanism design and not novel. (Mechanisms designed by Pafnuty Chebyshev in the 19th century are still widely used in locomotion.)

On the other hand, morphology may also refer to the dynamic properties of the robot, such as mass distribution and compliance. This leads to *dynamical* morphological computation, which not only simplified control but can also lead to high energy efficiency. After all, whereas in traditional control theory the dynamics of a system are cancelled out as much as possible, to steer the system's states to a desired point or along a desired trajectory, in dynamic morphological computation the inherent dynamics of the system *are* the desired dynamics, so the natural behaviour of the system *already* follows the desired trajectories. Successful examples of *dynamic* morphological computation may be found in Poulakakis (2006) or Iida et al. (2005), quadrupedal robots with almost-passive locomotion behaviour.

One possible approach to morphological computation is *biomimetic* morphological computation, where "proper" morphology is observed in nature and translated to morphological concepts for robots, as in Full and Koditschek (1999). (Parts of) the field of Soft Robotics may also be considered as morphological computation; see Kim et al. (2013). At the

same time, an attempt is made at a proper theoretical foundation for morphological computation, viz. Hauser et al. (2011) and Füchslin et al. (2012).

In dynamic morphological computation, Port-Hamiltonian System theory and energy-based robotics can make an important contribution. After all, morphological computation is concerned with properly interconnected physical storage elements (masses, springs). In Port-Hamiltonian System theory, those storage elements and their interconnection are *explicitly* modelled by their energy functions and the Dirac structure or internal interconnection matrix (J). Therefore, modelling the combination of system and controller as interconnected PHSs gives direct insight into the dynamic morphology. Furthermore, the distinction between "controller" and "system" completely disappears, since both are described, modelled and designed as a PHS. And if the system's inherent, natural dynamics exhibit desired behaviour, the control system merely needs to inject energy to keep the motion going and compensate for e.g. friction losses, and possibly stabilise the oscillations.

The energy-based modelling and control approach is a perfect match for designing, analysing and understanding morphological computation. The application of Port-Hamiltonian System theory and energy-based robotics to the field of morphological computation was first studied in Folkertsma et al. (2015), where a fully elastic spine was shown to generate locomotion in a quadruped, with control limited to energy injection through the 1-DoF legs.

5

Conclusion

Robots, like all physical systems, are governed by energy. Interaction of the actuators with the mechanical system of the robot, and interaction of the robot with the environment, is characterised by energy exchange. Describing systems and controllers in an energy-based way gives direct insight into these energy flows in the physical system. In this work, we have shown how energy-based theories and tools can be applied to the field of robotics.

We have used the Port-Hamiltonian System theory to model the robot with energy at the centre. We have shown methods to measure and control energy flows in practical systems. Energy-based thinking can also be used for controller design: whether in control by interconnection, where the control algorithms are "virtual physical systems" in Port-Hamiltonian fashion; or in control by *physical* interconnection, where the mechanical design of the robot is based on the desired energetic behaviour.

This energy-based view of the mechanical system, actuators and control algorithms dissolves the traditional boundaries between them: it is irrelevant whether a behaviour is the result of a physical process, or of a virtual process. It is a holistic, energy-based view of the robot.

The focus of this work has been on cyber-physical interaction: on the flow of energy between the physical robot and the virtual controller. In places, we have made the connection to classical control theory: the (energy-based) impedance control equivalence of PD control (§3.1.3); energy shaping methods (§3.2); Port-Hamiltonian System analysis of Variable Stiffness Actuators (§4.2); but also augmenting arbitrary control and actuation systems with energy awareness or energy budgets (§2.3). Once the total Hamiltonian is known, Lyapunov's method can be used for finding stabilising energy-based control laws (§A).

An important application of "Energy in Robotics" that was not studied is the use of energy in navigation. Potential-field based methods, for example, construct virtual energy fields that create barriers around objects (object avoidance) or make the robot "flow" naturally towards a destination (navigation). This is in essence also a form of energy shaping. Furthermore, we have restricted ourselves to simple examples with straightforward control implementations and clear simulation examples. This clarity has hopefully piqued interest in applying energy-based method to robotics: different examples, real-life applications and more complex results can be found in the references.

Acknowledgements

These lecture notes are a compendium of research work done by ourselves and others, independently and in collaboration. Credits go to all those researchers who have worked on topics of energy and robotics. The authors specifically want to thank Arjan van der Schaft, for a long-time collaboration, technical discussions and inspiring talks.

In our lab, we apply these energy-based concepts to robot design and controller architectures. That is to say, there are a large number of students, PhD students, postdocs, technicians and other colleagues who have done this work, or made it possible. Thanks, to all of you who contributed directly or indirectly to this work.

Appendices

A

Energy control: proof

The mass-spring system with force actuator, described in (3.19), is interconnected with the passive controller of the form (3.20). If g is chosen as derived in (3.21)–(3.22), the system is exponentially stable at desired energy level $E_\mathrm{s} = \frac{k}{2}x_\mathrm{s}^2$.

Proof. The closed-loop dynamics of the interconnected system are the following:

$$\mathcal{H}_\mathrm{tot}(x) = \frac{p^2}{2\mathrm{m}} + \frac{\mathrm{k}}{2}x_\mathrm{m}^2 + \frac{q^2}{2\mathrm{C}} \tag{A.1}$$
$$\begin{pmatrix} \dot{p} \\ \dot{x}_\mathrm{m} \\ \dot{q} \end{pmatrix} = \begin{bmatrix} 0 & -1 & g \\ 1 & 0 & 0 \\ -g & 0 & 0 \end{bmatrix} \begin{pmatrix} p/m \\ \mathrm{k}x_\mathrm{m} \\ q/\mathrm{C} \end{pmatrix},$$

with g chosen as

$$g := -\frac{\lambda}{2}\frac{\mathrm{C}}{qv}(E - E_\mathrm{s}); \quad E := \frac{p^2}{2\mathrm{m}} + \frac{\mathrm{k}}{2}x_\mathrm{m}^2. \tag{A.2}$$

Note that there are two issues with this proposed control law, namely: $v = 0$ or $q = 0$. The latter case corresponds to an empty energy buffer in the controller, which must be avoided—if more energy needs to be

injected to reach E_s than is available, it will of course never reach E_s. On the contrary, the first case happens often during the oscillations. In practice, g should be set to some constant, e.g. 0 or 1, whenever $v = 0$ (or $v < \epsilon$, some small threshold).

Energy-like functions are often chosen as Lyapunov candidate functions. In this case, we can use the actual energy error $E - E_\mathrm{s}$:

$$V := \frac{1}{2}(E - E_\mathrm{s})^2, \tag{A.3}$$

which is radially unbounded and has a single minimum 0 at $E = E_\mathrm{s}$. The time derivative of V is derived in (A.4)–(A.9) below. To obtain (A.6), the system dynamics of (A.1) are used; in (A.7) the control law was used and p/m substituted for v, again from the system dynamics.

$$\begin{aligned}
\frac{\mathrm{d}V}{\mathrm{d}t} &= (E - E_\mathrm{s})\left(\frac{\mathrm{d}}{\mathrm{d}t}E\right) && \text{(A.4)}\\
&= (E - E_\mathrm{s})\left(\frac{p}{\mathrm{m}}\dot{p} + \mathrm{k}x_\mathrm{m}\dot{x}_\mathrm{m}\right) && \text{(A.5)}\\
&= (E - E_\mathrm{s})\left(\frac{p}{\mathrm{m}}(-\mathrm{k}x_\mathrm{m} + gq/\mathrm{C}) + \mathrm{k}x_\mathrm{m}(p/\mathrm{m})\right) && \text{(A.6)}\\
&= (E - E_\mathrm{s})\left(\frac{pq}{\mathrm{mC}}\left(-\frac{\lambda}{2}\frac{\mathrm{C}}{qv}(E - E_\mathrm{s})\right)\right) && \text{(A.7)}\\
&= (E - E_\mathrm{s})\left(-\frac{\lambda}{2}(E - E_\mathrm{s})\right) && \text{(A.8)}\\
&= -\lambda V. && \text{(A.9)}
\end{aligned}$$

This derivation shows that $\dot{V}$ is strictly negative when $V \neq 0$, except when $p = 0$: then g is set to some constant and (A.6) will be 0. From LaSalle's invariance principle, the final trajectories of the system will all lie in the set $\left\{x(t)|\dot{V}(x) = 0\right\}$, which is the union of $\{x(t)|V(x) = 0\}$ and $\{x(t)|p = 0\}$. The first set is the trajectory where $E = E_\mathrm{s}$; the second set is only an invariant set when $p = 0 \wedge \dot{p} = 0$, i.e. it contains only the trivial trajectory $x_\mathrm{m}(t) = 0$.

It can be concluded that V decreases exponentially fast with λ and hence the system is globally exponentially stable at $E = E_\mathrm{s}$, if there is sufficient energy in the control buffer such that $q > 0 \forall t$. □

References

P. C. Breedveld. Modeling and simulation of dynamic systems using bond graphs. *Encyclopedy of Life Support Systems (EOLSS)*, pages 1–37, 2008.

M. Kanat Camlibel, A. Agung Julius, Ramkrishna Pasumarthy, and Jacquelien M. A. Scherpen, editors. *Mathematical Control Theory I: Nonlinear and Hybrid Control Systems*, volume 461 of *Lecture Notes in Control and Information Sciences*. Springer International Publishing, 1st edition, 2015. ISBN 978-3-319-20987-6.

J. Cervera, Arjan J. van der Schaft, and A. Baños. Interconnection of port-Hamiltonian systems and composition of Dirac structures. *Automatica*, 43 (2):212–225, Feb 2007. ISSN 00051098.

Douwe Dresscher. *Controlled Passive Actuation: concepts for energy efficient actuation using mechanical storage elements and continuously variable transmissions*. PhD thesis, University of Twente, Robotics and Mechatronics group, 2016.

Vincent Duindam and Stefano Stramigioli. Port-based asymptotic curve tracking for mechanical systems. *European Journal of Control*, 10(5):411–420, 2004.

Vincent Duindam, Alessandro Macchelli, Stefano Stramigioli, and Herman Bruyninckx. *Modeling and Control of Complex Physical Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-642-03195-3.

Gerrit A. Folkertsma, Sangbae Kim, and Stefano Stramigioli. Parallel stiffness in a bounding quadruped with flexible spine. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2210–2215, Oct 2012.

Gerrit A. Folkertsma, Arjan J. van der Schaft, and Stefano Stramigioli. Power-continuous synchronisation of oscillators: A novel, energy-free way to synchronise dynamical systems. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1493–1498, May 2014.

Gerrit A. Folkertsma, Arjan J. van der Schaft, and Stefano Stramigioli. Morphological computation in a fast-running quadruped with elastic spine. In *Proceedings of Lagrangian and Hamiltonian Methods for Nonlinear Control (LHMNLC15)*, July 2015.

M. C. J. Franken, Stefano Stramigioli, Rob Reilink, Cristian Secchi, and A. Macchelli. Bridging the gap between passivity and transparency. In *Proceedings of Robotics: Science and Systems*, page 36, Seattle, USA, June 2009.

Rudolf M. Füchslin, Andrej Dzyakanchuk, Dandolo Flumini, Helmut Hauser, Kenneth J. Hunt, Rolf H. Luchsinger, Benedikt Reller, Stephan Scheidegger, and Richard Walker. Morphological computation and morphological control: Steps toward a formal theory and applications. *Artificial Life*, 19(1):9–34, Nov 2012. ISSN 1064-5462.

R. J. Full and D. E. Koditschek. Templates and anchors: Neuromechanical hypotheses of legged locomotion on land. *The Journal of Experimental Biology*, 202(Pt 23):3325–3332, Dec 1999. ISSN 0022-0949.

V. R. Ham, T. G. Sugar, B. Vanderborght, K. W. Hollander, and D. Lefeber. Compliant actuator designs: Review of actuators with passive adjustable compliance/controllable stiffness for robotic applications. *IEEE Robotics and Automation Magazine*, 16(3):81–94, 2009.

Helmut Hauser, Auke J. Ijspeert, Rudolf M. Füchslin, Rolf Pfeifer, and Wolfgang Maass. Towards a theoretical foundation for morphological computation with compliant bodies. *Biological Cybernetics*, 105(5):355–370, 2011. ISSN 1432-0770.

Just L. Herder. Design of spring force compensation systems. *Mechanism and Machine Theory*, 33(1):151–161, January 1998.

Neville Hogan. Impedance control: An approach to manipulation. *Journal of Dynamic Systems, Measurement and Control*, 107(1):1–24, March 1985.

J. Hunt, F. Giardina, A. Rosendo, and Fumiya Iida. Improving efficiency for an open-loop-controlled locomotion with a pulsed actuation. *IEEE/ASME Transactions on Mechatronics*, 21(3):1581–1591, June 2016. ISSN 1083-4435.

Fumiya Iida, G. Gómez, and Rolf Pfeifer. Exploiting body dynamics for controlling a running quadruped robot. In *Proceedings of the 12th International Conference on Advanced Robotics, 2005. ICAR'05*, pages 229–235, 2005. ISBN 0780391772.

Theo Jansen. Strandbeest, nov 2016. URL http://strandbeest.com.

Sangbae Kim, Cecilia Laschi, and Barry Trimmer. Soft robotics: a bioinspired evolution in robotics. *Trends in Biotechnology*, 31(5):287–294, 2013. ISSN 0167-7799.

P. Y. Li and R. Horowitz. Passive velocity field control of mechanical manipulators. *IEEE Transactions on Robotics and Automation*, 15(4):751–763, Aug 1999. ISSN 1042-296X.

S. M. Lukic, J. Cao, R. C. Bansal, F. Rodriguez, and A. Emadi. Energy storage systems for automotive applications. *IEEE Transactions on Industrial Electronics*, 55(6):2258–2267, 2008.

Romeo Ortega, A. Loria, R. Kelly, and L. Praly. On passivity-based output feedback global stabilization of euler-lagrange systems. In *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, volume 1, pages 381–386, Dec 1994.

Romeo Ortega, A. J. van der Schaft, Iven Mareels, and Bernard Maschke. Putting energy back in control. *IEEE Control Systems*, 21(2):18–33, Apr 2001. ISSN 1066-033X.

Romeo Ortega, Arjan van der Schaft, Fernando Castaños, and Alessandro Astolfi. Control by Interconnection and Standard Passivity-Based Control of Port-Hamiltonian Systems. *IEEE Trans. Automat. Contr.*, 53(11):2527–2542, 2008.

H. M. Paynter. *Analysis and design of engineering systems*. MIT Press, MA, 1960.

Rolf Pfeifer, Max Lungarella, and Fumiya Iida. Self-organization, embodiment, and biologically inspired robotics. *Science*, 318(5853):1088–1093, 2007.

M. Plooij and M. Wisse. A novel spring mechanism to reduce energy consumption of robotic arms. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2901–2908, Oct 2012.

I. Poulakakis. On the stability of the passive dynamics of quadrupedal running with a bounding gait. *The International Journal of Robotics Research*, 25 (7):669–687, July 2006. ISSN 0278-3649.

G. A. Pratt and M. M. Williamson. Series elastic actuators. In *Intelligent Robots and Systems 95. 'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on*, volume 1, pages 399–406, Aug 1995.

A. Sanchez-Squella, R. Ortega, R. Grino, and S. Malo. Dynamic energy router. *IEEE Control Systems*, 30(6):72–80, Dec 2010. ISSN 1066-033X.

C. Secchi, S. Stramigioli, and C. Fantuzzi. Dealing with unreliabilities in digital passive geometric telemanipulation. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, volume 3, pages 2823–2828, Oct 2003.

S. Seok, A. Wang, M. Y. (Michael) Chuah, D. J. Hyun, J. Lee, D. M. Otten, J. H. Lang, and S. Kim. Design principles for energy-efficient legged locomotion and implementation on the mit cheetah robot. *IEEE/ASME Transactions on Mechatronics*, 20(3):1117–1129, June 2015. ISSN 1083-4435.

Meera Sitharam and Menghan Wang. How the beast really moves: Cayley analysis of mechanism realization spaces using caymos. *Computer-Aided Design*, 46:205–210, 2014. ISSN 0010-4485. 2013 SIAM Conference on Geometric and Physical Modeling.

Stefano Stramigioli, C. Secchi, Arjan J. van der Schaft, and C. Fantuzzi. A novel theory for sampled data system passivity. In *IEEE/RSJ International Conference on Intelligent Robots and System, 2002*, volume 2, pages 1936–1941. IEEE, 2002.

Stefano Stramigioli, C. Secchi, Arjan J. van der Schaft, and C. Fantuzzi. Sampled data systems passivity and discrete port-Hamiltonian systems. *IEEE Transactions on Robotics*, 21(4):574–587, Aug 2005. ISSN 1552-3098.

Stefano Stramigioli. Creating artificial damping by means of damping injection. In K.Danai, editor, *Proceedings of the ASME Dynamic Systems and Control Division*, volume DSC.58, pages 601–606, Atlanta, (GE), 1996.

Stefano Stramigioli and M. Dijk. Energy Conservative Limit Cycle Oscillations. In *17th IFAC World Congress*, pages 15666–15671, 2008. ISBN 9781123478.

Stefano Stramigioli, Bernhard Maschke, and Arjan J. van der Schaft. Passive output feedback and port interconnection. In *Proceedings of 4th IFAC NOLCOS*, pages 613–618, Enschede, Netherlands, 1998.

Stefano Stramigioli, Ernie Fasse, and Jan C. Willems. A rigorous framework for interactive robot control. *International Journal of Control*, 75(18): 1486–1502, 2002.

T. S. Tadele, T. de Vries, and Stefano Stramigioli. The safety of domestic robotics: A survey of various safety-related publications. *Robotics Automation Magazine, IEEE*, 21(3):134–142, Sept 2014. ISSN 1070-9932.

Arjan J. van der Schaft. *L2-Gain and Passivity Techniques in Nonlinear Control*. Communications and Control Engineering. Springer International Publishing, 3rd edition, 2017. ISBN 978-3-319-49991-8.

Arjan J. 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. ISSN 2325-6818.

B. Vanderborght, R. Van Ham, D. Lefeber, T. G. Sugar, and K. W. Hollander. Comparison of mechanical design and energy consumption of adaptable, passive-compliant actuators. *International Journal of Robotics Research*, 28 (1):90–103, 2009.

B. Vanderborght, A. Albu-Schaeffer, A. Bicchi, E. Burdet, D. G. Caldwell, R. Carloni, M. Catalano, O. Eiberger, W. Friedl, G. Ganesh, M. Garabini, M. Grebenstein, G. Grioli, S. Haddadin, H. Hoppner, A. Jafari, M. Laffranchi, D. Lefeber, F. Petit, S. Stramigioli, N. Tsagarakis, M. Van Damme, R. Van Ham, L. C. Visser, and S. Wolf. Variable impedance actuators: A review. *Robotics and Autonomous Systems*, 61(12):1601–1614, 2013.

Ludo C. Visser, R. Carloni, and S. Stramigioli. Energy-efficient variable stiffness actuators. *IEEE Transactions on Robotics*, 27(5):865–875, Oct 2011a. ISSN 1552-3098.

Ludo C. Visser, Stefano Stramigioli, and Antonio Bicchi. Embodying desired behavior in variable stiffness actuators. *{IFAC} Proceedings Volumes*, 44 (1):9733–9738, 2011b. ISSN 1474-6670. 18th {IFAC} World Congress.

Richard Volpe and Pradeep Khosla. The equivalence of second-order impedance control and proportional gain explicit force control. *The International Journal of Robotics Research*, 14(6):574–589, 1995.

I. Wanders, G. A. Folkertsma, and S. Stramigioli. Design and analysis of an optimal hopper for use in resonance-based locomotion. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5197–5202, May 2015.

Jan C. Willems. Dissipative dynamical systems part I: General theory. *Archive for Rational Mechanics and Analysis*, 45(5):321–351, 1972.

S. Wolf, G. Grioli, O. Eiberger, W. Friedl, M. Grebenstein, H. Höppner, E. Burdet, D. G. Caldwell, R. Carloni, M. G. Catalano, D. Lefeber, S. Stramigioli, N. Tsagarakis, M. Van Damme, R. Van Ham, B. Vanderborght, L. C. Visser, A. Bicchi, and A. Albu-Schäffer. Variable stiffness actuators: Review on design and components. *IEEE/ASME Transactions on Mechatronics*, 21(5): 2418–2430, Oct 2016. ISSN 1083-4435.

Justin Won, Stefano Stramigioli, and Neville Hogan. Comment on "the equivalence of second-order impedance control and proportional gain explicit force control". *The International Journal of Robotics Research*, 16(6):873–875, 1997.