Control of a multi-robot cooperative team guided by a human-operator

Zwischenbericht zur
MASTERARBEIT
von

cand. ing. Martin Angerer

geb. am 10.06.1991
wohnhaft in:
Steinheilstrasse 5
80333 München
Tel.: 0151 57978548

Lehrstuhl für
INFORMATIONSTECHNISCHE REGELUNG
Technische Universität München

Univ.-Prof. Dr.-Ing. Sandra Hirche

Betreuer: Selma Musić, M.Sc.
Beginn: 01.10.2015
Zwischenbericht: 21.01.2016
Abgabe: 01.04.2016

In your final hardback copy, replace this page with the signed exercise sheet.

Abstract

Cooperative manipulation with human guidance can be used to solve versatile tasks. A new approach to system modelling and control in cooperative manipulation is the use of port-Hamiltonian systems. Starting from modelling of a cooperative manipulation set-up a model-based controller in the framework of Intrinsically Passive Control is derived. In contrast to the quasi-static implementations for robot hands, the controller has fully dynamic impedance relations. The good dynamic performance of the proposed control scheme is shown by simulation and compared to simulations of state-of-the-art controllers in cooperative manipulation.

Zusammenfassung

Hier die deutschsprachige Zusammenfassung

Contents

1 Introduction **5**
1.1 Problem Statement 7
1.2 Related Work 7

2 port-Hamiltonian system modelling **11**
2.1 Hamiltonian description of mechanical systems 11
2.1.1 Dirac structures and interconnection ports 14
2.2 3D-space modelling of mechanical systems 16
2.2.1 Euclidean space and motions 16
2.2.2 Dynamics of physical components 18
2.3 Imposing constraints 22
2.4 Spring-mass-damper systems 28

3 port-Hamiltonian based control **31**
3.1 Control architecture 31
3.2 Human guidance 34
3.3 Performance Comparison of control strategies 35
3.3.1 Internal impedance control with feed-forward of the object dynamics 35
3.3.2 Internal and external impedance based reference trajectory generation 36
3.3.3 Intrinsically Passive Controller (IPC) 37

4 Conclusion **39**

List of Figures **41**

Bibliography **43**

Chapter 1

Introduction

Cooperative manipulation involves two or more robot arms cooperatively moving or manipulating a common object [CM08]. Such a set-up overcomes many limitations seen by single robots. The prime example is the transportation of a large or heavy object, where a single manipulator would exert large local wrenches on the object. Back in the 1940s two remote manipulators were used to handle radioactive goods [Goe52]. Around 1990 the *NASA* researched cooperative manipulation in space construction applications [SC92]. Specifically in [LS05] a cooperative manipulation strategy for the repair of the Hubble Space Telescope is proposed. Other tasks of cooperative manipulation are for example:

- assembly of multiple parts without using special fixtures
- grasping an object without rigid fixture but by exerting a suitable squeezing force
- deforming a flexible object
- coordinated use of tools

A recent application has been presented at the Fukushima Daiichi Nuclear Power Station. By equipping a two-armed robot with different tools the robot is capable of grasping an item (e.g. piping) and cutting it with the other, see Fig. 1.1. For sensitive tools like an angle grinder it is favourable that movement and contact forces of both arms are coordinated.

Space and hazardous environments are fruitful environments for cooperative manipulators. Often they are controlled by a human operator from a safe location. Combining human reasoning and the enhanced flexibility of a cooperative set-up is a powerful combination in an unstructured environment. A human in the control loop comes with superior foresight and planning capabilities. To fully exploit this additional flexibility we require control architectures able to perform all the aforementioned tasks and operable in a comfortable and efficient manner.

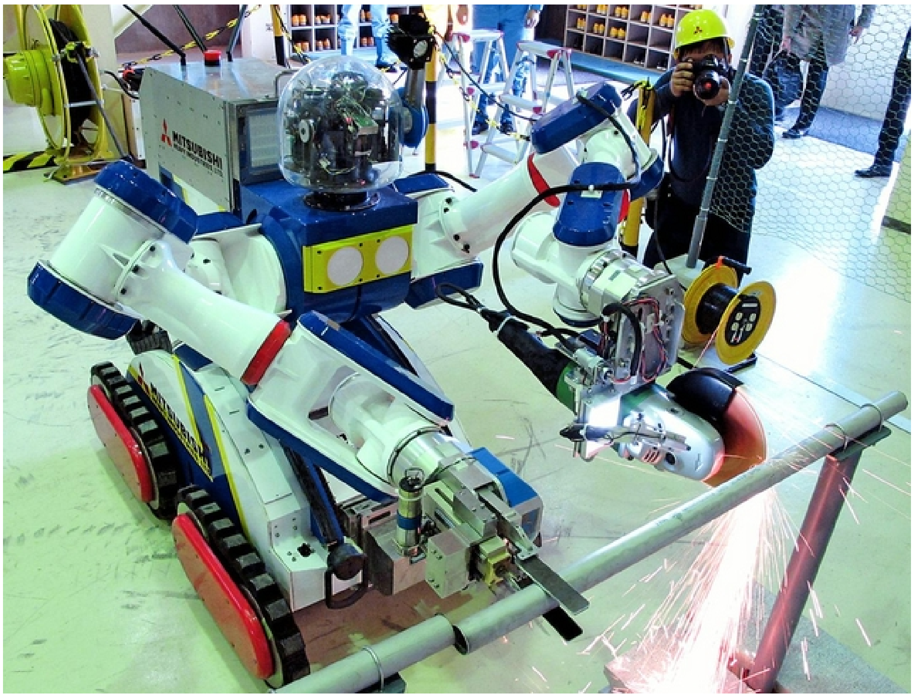

Figure 1.1: Demonstration of the tele-operated MHI MEISTeR robot at Fukushima Daiichi Nuclear Power Station [LTD14]

To address the first requirement the controller realizes changes of formation and relocates the constrained system. A change of formation first of all means the opening and closing of a grasp a around a common object. A virtual object and variable rest-length springs are introduced, the springs couple the virtual and the actual object. The size of virtual object specifies the rest-lengths, by choosing it smaller than the actual a grasping force is exerted. By choosing it bigger the grasp is opened and the actual object is released. Changing of rest-lengths means to change the potential energy stored in the springs. Since the controller itself is passive, the energy is exchanged with a distinct power port. In order to relocate the formation a spring is connected from the virtual object center to the desired new position/orientation. This ensures compliant behavior between (virtual) object and environment. Note that this concept is called *Intrinsically Passive Controller* (IPC) and was introduced by Stramigioli [Str01b], the control architecture described in the main part will closely stick to this concept.
The controller provides a reasonable layer of abstraction and takes responsibilities from the operator, in order to not overstrain her/his attention and let him focus on important elements of the task. To ensure a secure and stable grasp, the formation of robots is locally controlled without the help of the human. For dynamic manipulation tasks grasp force optimization can be utilized to limit internal to the absolute minimum. The control system provides appropriate feedback to provide better perception on the work environment and to make the job more intuitive. The user

specifies system motion and is provided with information about the necessary forces to accomplish the commanded motion. Thereby s/he gets a natural intuition of how much work has been done by the robots. To this end the controller shall never be a source of additional energy, the only way energy is fed into the system is by request of the operator, i.e. the controlled robots are passive. The maximum extractable energy is always bounded, the operator can estimate the stored energy and possible effects when interacting with the environment. Passive robotic systems are always stable when interconnected passive environments/humans and can be stable with some active systems. On the contrary, if the controlled robot is not passive, there is always a passive environment that destabilizes the interconnected system [Str15].

1.1 Problem Statement

Manipulating an unknown object in an unstructured environment is an interesting task for a human-guided cooperative manipulation set-up. Dropping the assumption of a rigid- connection between manipulator and object, the grasp has to be actively stabilized under varying circumstances. During dynamic manipulation a slipping of contact due to the inertia of the object has to be avoided using an automatic mechanism. On the other hand the operator must directly adjust grasp forces for heavy or fragile objects.
Position and velocity of the manipulated object are difficult to track in an everyday setting. Usually only end-effector position and velocity of the robots are available. Object position has to be estimated from the known data.
For grasping an object of unknown or even flexible shape, the determination of a fixed grasp map in advance is not useful. Size and shape of the grasp have to be determined by the operator during the grasp process. The controller has to be flexible to varying grasp geometries during the whole task execution.
Integrating the human into the control loop, energetic passivity of the closed loop system is a meaningful and intuitive way to ensure stability and a natural way of interaction. Energetic passivity limits the extractable mechanical energy from the closed loop system. This means that the potential damage is also limited. Energetic passivity of the robotic system ensures stability of the interconnection stability with any passive systems. Since humans and many relevant environments are passive the closed loop system will always be stable if the overall control scheme is designed energetically passive.

1.2 Related Work

A notable class of control architectures for cooperative manipulation is hybrid position/force control, with a motion control loop for trajectory tracking and a force control loop for internal forces [WKD92, Hsu93]. Their drawback is the inability to handle non-contact to contact transitions. Hogan introduced impedance control, which enforces a relation between force and motion [Hog84]. Its first application in cooperative manipulation was in [SC92] for realizing compliant object-environment interaction (external impedance control). Bonitz and Hsia [BH96] applied the concept to the manipulator-object relation (internal impedance control).

More recent impedance control schemes can be classified in terms of the information and sensor data available for the problem. Frugal architectures are formation control [SMH15] and the static IPC [WOH06]. Both do not incorporate the object dynamics in control, thus very little knowledge about the object (e.g. dimensions) is necessary. The control loops depend only on relative positions and velocities of the manipulators and do not require object tracking. Neglecting considerable object dynamics is an obvious drawback of the schemes.

The concept of the *Intrinsically Passive Controller* (IPC), introduced by Stramigioli [Str01b] and called dynamic IPC by Wimböck et al. [WOH08], tries to overcome some of the limitations. In the controller the object is represented by a virtual pendent and simulated to reproduce its dynamics for control purpose. This has the advantage that still no tracking of the object is required, object velocity and even acceleration can be obtained from the simulation.

Techniques which rely on exact knowledge of the object motion were introduced by Caccavale et al. [CV01, CCMV08] and more recent [HKDN13] and [DPEZ+15]. It is common to them that they assume rigid fixtures to a rigid object, these conditions overcome the problem of object tracking. The approaches by Caccavale and coworkers use force/torque sensors at the manipulators to establish compliant object environment interaction, for this purpose Heck et al. [HKDN13] assume to have an exact model of the environment. Stramigioli's IPC implements a compliant relation between virtual object and environment.

The human operator must be able control the manipulators at a reasonable degree of complexity, therefore in an direct master-slave approach each robot is controlled independently by a human operator. Exactly coordinating their motions is a difficult task for humans, as a consequence a certain amount of autonomy is left to the robot system, enabling a single operator to control the cooperative system. Lee and Spong [LS05] apply the master-slave scheme but treat the constrained system as a single slave, while the formation is preserved by the robots autonomously. Many master-slave systems give the operator force-feedback, while s/he commands the motion. This helps the operator compensate for resistances and gives a natural feeling of the interaction with the environment. The structure then is fully bi-directional, ones refers to bilateral telemanipulation [NPH08]. Leader-follower [SMH15, SMP14] schemes differ in terms of feedback provided to the operator. Tactile and visual types are non-reactive, i.e. they do not induce operator movements reacting to a back-driving force [MT93]. A purely vision-based architecture is introduced by Gioioso et al. [GFS+14], hand gestures are used to both control the motion of the constrained system and the opening and closing of a grasp. Control architectures that leave even further autonomy to the robot system and possess a closed local, autonomous control loop, are categorized as supervisory control. They interacts with the opera-

tor by continuously sending information about the state and periodically receiving commands [She92].
The assumption of rigid fixtures between manipulators and object is very common in cooperative manipulation, Lee and Spong [LS05] are an exception. Friction grasps are mainly researched in robot hand literature ([WOH06, WOH08, Str01b]). For the stabilization of a friction grasp it is vital to choose appropriate forces depending on the dynamic state. Therefore the Coulomb friction constraints along with other criteria (safety margins, force limits) can be formulated as a cost function for optimization. Buss et al. [BHM96] realized that the Coulomb friction constraints can be formulated as positive definite matrices, Han et al. [HTL00] gave a linear matrix inequality problem. For this type of optimization problems very efficient, real-time solvers exist.

Chapter 2

port-Hamiltonian system modelling

In this chapter a modelling approach of interactive mechanical systems based on the port-Hamiltonian framework is introduced. Port based modelling aims at providing a consistent system description over different physical domains. In the presented case we stick to the mechanical domain, with two energies types of interest: kinetic and potential energy. Modelling is done by dividing the system in simple Hamiltonian subsystems and interconnecting them using methods of network theory.
Kinematic constraints limit the possible motion of the system's sub-members and impose a set of differential-algebraic equations (DAE). The DAE description is reformulated into an input-output representation, which is beneficial when it comes to control.

2.1 Hamiltonian description of mechanical systems

The Hamiltonian equations of motion are related to the better known Euler-Lagrange equations with the Legendre transformation. Both are formulations on energy level thus they are useful for a continuous description of mixed-energy systems, e.g. a pendulum alternating between kinetic and potential energy periodically. The classical Hamiltonian equations for a mechanical system are:

$$\begin{aligned} \dot{q} &= \frac{\partial H}{\partial p}(q,p) \\ \dot{p} &= -\frac{\partial H}{\partial q}(q,p) + F \end{aligned} \tag{2.1}$$

Where Hamiltonian $H(q,p)$ is the total energy of the system, $q = (q_1, ..., q_k)^T$ denotes the generalized coordinates of the system with k-degrees of freedom and $p = (p_1, ..., p_k)^T$ is the vector of the generalized momenta. F is the input of generalized forces.

An important aspect of PHS is conservation of energy, i.e. supplied work is stored in the system [vdS06]. The energy balance of the system is:

$$\frac{d}{dt}H = \frac{\partial^T H}{\partial q}(q,p)\dot{q} + \frac{\partial^T H}{\partial p}(q,p)\dot{p} = \frac{\partial^T H}{\partial p}(q,p)F = \dot{q}^T F \tag{2.2}$$

Note that the product of generalized force and velocity is mechanical power. Depending on the energy type of the mechanical system (kinetic or potential), the input is either or force or velocity and the output being velocity or force respectively. This motivates the more general concept of flows f and efforts e, being the input and output of a pHs.

$$\begin{aligned} \dot{q} &= \frac{\partial H}{\partial p}(q,p) \\ \dot{p} &= -\frac{\partial H}{\partial q}(q,p) + g(q)f \\ e &= g^T(q)\frac{\partial H}{\partial p}(q,p) \end{aligned} \tag{2.3}$$

Example: Consider a simple one-dimensional spring-mass system described by $m\ddot{x} = -kx + F_e$. Where m, k, F denote the mass, stiffness and external force acting on the mass. We can give a state space formulation for the system

$$\begin{pmatrix}\dot{x}\\ \ddot{x}\end{pmatrix} = \begin{pmatrix}0 & 1\\ -\frac{k}{m} & 0\end{pmatrix}\begin{pmatrix}x\\ \dot{x}\end{pmatrix} + \begin{pmatrix}0\\1\end{pmatrix}\frac{F_e}{m} \tag{2.4}$$

In the Hamiltonian approach the system is described based on the Hamiltonian energy functions, being $V_s = \frac{1}{2}kq^2$ for the spring and $V_m = \frac{1}{2m}p^2$ for the mass. The state variables are thus replaced by the energy states, the *configuration* $q = x$ accounting for the spring and the *momentum* $p = m\dot{x}$ accounting for the mass.

$$\begin{aligned} \begin{pmatrix}\dot{q}\\ \dot{p}\end{pmatrix} &= \begin{pmatrix}0 & 1\\ -1 & 0\end{pmatrix}\begin{pmatrix}\frac{\partial V}{\partial q}(q,p)\\ \frac{\partial V}{\partial p}(q,p)\end{pmatrix} + \begin{pmatrix}0\\1\end{pmatrix}F_e \\ e &= \begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix}\frac{\partial V}{\partial q}(q,p)\\ \frac{\partial V}{\partial p}(q,p)\end{pmatrix}, \; V = V_s + V_m \end{aligned} \tag{2.5}$$

Key aspect of pHs is the division into atomic energy storing elements (spring,mass) and their proper interconnection. In the example we have implicitly done this by defining energy functions and states for both elements. The rules of interconnection are explicitly given by: :

- equal velocity of spring tip and mass $\dot{x} = \frac{\partial V}{\partial p}$(rigid connection)
- opposite forces at spring tip and mass $\dot{p} = -\frac{\partial V}{\partial q} + F_e$ (principle of action-reaction)

Table 2.1: PHS variables of mechanical elements

	Spring	**Mass**	**Damper**
Effort variable	Force F	Velocity $\dot{x}$	Force F
Flow variable	Velocity $\dot{x}$	Force $F = \dot{p}$	Velocity $\dot{x}$
State variable	Position x	Momentum p	-
Energy function	$E(x) = \frac{1}{2}kx^2$	$E(p) = \frac{p^2}{2m}$	$E(\dot{x}) = D\dot{x}^2$ (diss. co-energy)

Due to the *second law of thermodynamics* real mechanical systems are never energy conservative. Thus we require energy dissipating elements (since thermal energy is "lost" w.r.t. the mechanical domain). A mechanical system is thus described by its basic elements: springs, masses and dampers. Table 2.1 gives an overview of the elements and their characterizing quantities. Note that dissipation elements do not have a state because they are static. It is worth pointing out that the *flows* are the time derivatives of the *states*.

Next to *energy-storing* and *-dissipating* there is a third class of elements, namely *energy-conservative* structures. Elements within this group are transformers, gyrators and ideal constraints, they are used to redirect the power flow in the system. It is possible to merge all *energy-storing* elements into a single object representation (see Fig. 2.1). Analogously this can be done for the dissipation elements. The interconnecting structure (denoted by $\mathcal{D}$ in Fig. 2.1), consisting of the *energy-conservative* elements, formalizes the energy routing and geometric dependencies of the system. A detailed explanation is given in the next subsection.

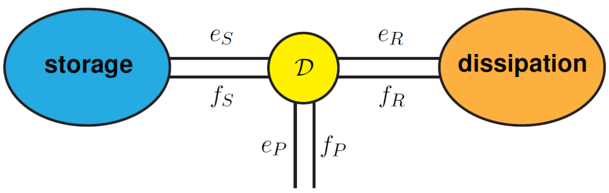

Figure 2.1: port-Hamiltonian system structure [vdS06]

Complex physical systems can be modelled as a network of energy storing and dissipating elements, similar to representation of electrical networks consisting of resistors, inductors and capacitors. The rules of interconnection are Newton's third law (action-reaction), Kirchhoff's laws and power-conserving elements like transformers or gyrators. The aim of PHS-modelling is to describe the power-conserving elements with the interconnection laws as a geometric structure and to define the Hamiltonian function as the total energy of the system.

The power flowing between the system's portions is described by the flow-effort product $e^T f$. A pair (f, e) is called a *port*. The external port (f_P, e_P) describes the energy flow to/from environment or a controller.

2.1.1 Dirac structures and interconnection ports

The energy-routing structure forms the basis of every pHs. It can be compared to the printed circuit board in electronics, while capacitors, inductors and resistors being the energy-storing and damping elements. Mathematically it has the form of a *Dirac* structure. The main property of a Dirac structure is power conservation, i.e. the power flowing into and out of it always sums to zero. We can define the set of ports (f, e) connecting to the Dirac structure $\mathcal{D}$, thus

$$e^T f = 0 \;\; \forall (f, e) \in \mathcal{D} \tag{2.6}$$

Where $\mathcal{D}$ is a subspace of the space of flow and effort $\mathcal{D} \subset \mathcal{E} \times \mathcal{F}$. The space of flows is $f \in \mathcal{F}$ the space of efforts is its dual linear space $e \in \mathcal{E} = \mathcal{F}^*$. The Dirac structure has the same dimension than the space of flow $dim\mathcal{D} = dim\mathcal{F}$. Further mathematical requirements can be found in literature [vdS06, vdSJ14]. The interconnection of ports is defined by the Dirac structure matrix D

$$\begin{pmatrix} e_P \\ f_S \\ e_R \end{pmatrix} = D \begin{pmatrix} f_P \\ e_S \\ e_R \end{pmatrix} \tag{2.7}$$

with the internal elements

$$D := \begin{pmatrix} D_P & G_1 & G_2 \\ -G_1^T & D_S & G_3 \\ -G_2^T & -G_3^T & D_R \end{pmatrix} \tag{2.8}$$

and D_P, D_S, D_R are skew-symmetric. Thus $D = -D^T$ is skew-symmetric, this is necessary for $e^T f = 0$. The power exchanged through a port is given by $e_*^T f_*$. In Fig. 2.1 the port variables entering the Dirac structure have already been split.

Energy storage port

The port accounts for the internal storage of the system, its port variables are (f_S, e_S). The power supplied through this port is stored in the Hamiltonian energy function $V(x)$ of the system. Here x denotes a general energy state variable. The resulting energy balance is:

$$\frac{d}{dt} H = \frac{\partial^T V}{\partial x}(x)\dot{x} \tag{2.9}$$

The flow variable is the energy rate $f_S = -\dot{x}$ and the effort variable is $e_S = \frac{\partial H}{\partial x}(x)$.

Energy dissipation port

The port corresponds to internal dissipation and can be used to model resistive elements. The port variables are described by the general resistive relation

$$F(f_R, e_R) = 0 \tag{2.10}$$

with the property $e_R^T f_R \leq 0$ (energy dissipation). An important special case is the input-output resistive relation $f_R = -F(e_R)$, for linear elements simply

$$f_R = -Re_R \ , \ R = R^T \geq 0 \tag{2.11}$$

For an uncontrolled system that does not interact with the environment, i.e. no energy exchange through the external port, the energy balance is:

$$\frac{dH}{dt} = -e_S^T f_S = e_R^T f_R \leq 0 \tag{2.12}$$

External port

The external port (f_P, e_P) can be further split into an environment *interaction* (f_I, e_I) and a *control* port (f_C, e_C), satisfying $e_P^T f_P = e_I^T f_I + e_C^T f_C$. The power balance of the whole system then is

$$e_S^T f_S + e_R^T f_R + e_I^T f_I + e_C^T f_C = 0 \tag{2.13}$$

or by using (2.9)

$$\frac{dH}{dt} = e_R^T f_R + e_I^T f_I + e_C^T f_C \tag{2.14}$$

Interconnection of port-Hamiltonian systems

It is important to notice that the interconnection of two pHs is again a pHs [vdSJ14]. Consider two general pHs ($i = 1, 2$) with open control and environment interaction ports:

$$\dot{x}_i = (J_i - R_i)\frac{\partial H_i}{\partial x_i} + (g_i^C \quad g_i^I)\begin{pmatrix} f_i^C \\ f_i^I \end{pmatrix} \tag{2.15}$$

$$\begin{pmatrix} e_i^C \\ e_i^I \end{pmatrix} = \begin{pmatrix} (g_i^C)^T \\ (g_i^I)^T \end{pmatrix} \frac{\partial H_i}{\partial x_i} \tag{2.16}$$

Where J_i, R_i are a skew-symmetric structure matrix and a positive-semi-definite symmetric dissipation matrix. $(g_i^C g_i^I)$ is a general input matrix. For notational convenience the usual dependencies on the states have been omitted. The control inputs and outputs are now connected by setting $f_1^C = e_2^C$ and $f_2^C = -e_1^C$. Note that the minus sign is necessary for power conservation. The power exchanged by the i-th system is $P_i = (e_i^C)^T f_i^C$, therefore the total exchanged energy fulfils $P_1 + P_2 = 0$. The resulting interconnected system has still the environment interaction ports open:

$$\dot{x} = (J - R)\frac{\partial H}{\partial x} + (g_1^I \quad g_2^I)\begin{pmatrix} f_1^C \\ f_2^I \end{pmatrix} \tag{2.17}$$

$$\begin{pmatrix} e_1^I \\ e_2^I \end{pmatrix} = \begin{pmatrix} (g_1^I)^T \\ (g_2^I)^T \end{pmatrix} \frac{\partial H}{\partial x} \tag{2.18}$$

Where $x = (x_1, x_2)^T$ and $H = H_1 + H_2$ is the sum of the two energies. The structure and dissipation matrix become:

$$J = \begin{pmatrix} J_1 & g_1^C (g_2^C)^T \\ -g_1^C (g_2^C)^T & J_2 \end{pmatrix}$$

,

$$R = \begin{pmatrix} R_1 & 0 \\ 0 & R_2 \end{pmatrix}$$

2.2 3D-space modelling of mechanical systems

2.2.1 Euclidean space and motions

Coordinate frames

A coordinate frame of the three-dimensional Euclidean space is a 4-tuple of the form $\Psi = (o, \hat{x}, \hat{y}, \hat{z})$. Where o is the three-dimensional vector of the origin and $\hat{x}, \hat{y}, \hat{z}$ are the linear independent, orthonormal coordinate vectors. Consider two coordinate frames Ψ_1, Ψ_2 which share the same origin but differ in orientation due to different choices of $\hat{x}_i, \hat{y}_i, \hat{z}_i,\ i = 1, 2$. The change of orientation from Ψ_i to Ψ_i is described by the rotation matrix R_i^j. The set of rotation matrices is called *special orthonormal* group ($SO(3)$) [Str01a] and is defined as:

$$SO(3) = \{R \in \mathbb{R}^{3\times 3} \mid R^{-1} = R^T, detR = 1\} \tag{2.19}$$

Usually coordinate frames are defined with respect to an inertial frame, and the coordinate vectors $\hat{x}, \hat{y}, \hat{z}$ are chosen equal for all frames, deviations of orientation are represented by a rotation matrix relative to the inertial frame. In general a change of coordinate frames from Ψ_i to Ψ_j can be expressed with the homogeneous matrix

$$H_i^j := \begin{pmatrix} R_i^j & p_i^j \\ 0_{1\times 3} & 1 \end{pmatrix}$$

Where $p_i^j = o_j - o_i$ denotes the changes of origins. A point $p^1 \in \mathbb{R}^3$ expressed in Ψ_i is cast into Ψ_j by

$$\begin{pmatrix} p^j \\ 1 \end{pmatrix} = H_i^j \begin{pmatrix} p^i \\ 1 \end{pmatrix} \tag{2.20}$$

. The inverse transformation H_j^i is given by

$$H_i^j = (H_i^j)^{-1} = \begin{pmatrix} (R_i^j)^T & -(R_i^j)^T p_i^j \\ 0_{1\times 3} & 1 \end{pmatrix}$$

and is still a homogeneous matrix. The set of homogeneous matrices is called the *special Euclidean* group:

$$SE(3) := \{\begin{pmatrix} R & p \\ 0 & 1 \end{pmatrix} \mid R \in SO((3), p \in \mathbb{R}^3\} \tag{2.21}$$

The $SE(3)$ is a matrix lie group, composed of the set of homogeneous matrices H_i^j and the matrix multiplication being the group operation. For more information on Lie groups see e.g. [Str01b].

Twists and wrenches

Consider any point p not moving in coordinate frame Ψ_i, i.e. $\dot{p}^i = 0$. If p is moving in another coordinate frame Ψ_j, the two frame move with respect to each other. The trajectory can be described as a function of time: $H_i^j(t) \in SE(3)$. By differentiating (2.20) one obtains

$$\begin{pmatrix} \dot{p}^j(t) \\ 1 \end{pmatrix} = \dot{H}_i^j(t) \begin{pmatrix} p^i \\ 1 \end{pmatrix}$$

$\dot{H}_i^j$ describes both motion and a change of the reference frame. A separated representation is

$$\begin{pmatrix} \dot{p}^j(t) \\ 1 \end{pmatrix} = \tilde{T}_i^{j,j} \left(H_i^j \begin{pmatrix} p^i \\ 1 \end{pmatrix} \right) \tag{2.22}$$

More formally speaking $\dot{H}_i^j$ is a tangential vector along the trajectory $H_i^j(t)$ and thus in the tangent space of the $SE(3)$: $\dot{H}_i^j \in T_{H_i^j} SE(3)$. The tangent space $T_{H_i^j} SE(3)$ depends on the relation of Ψ_i and Ψ_j. To obtain a representation of motion which is referenced to a coordinate frame, we can map $\dot{H}_i^j$ to the identity of the $SE(3)$. At the identity e of the $SE(3)$ the tangent space $T_e SE(3)$ has the structure of a Lie algebra. The Lie algebra of the $SE(3)$ is denoted by $\mathfrak{se}(3)$. This is done either by left or right translation, for a definition see [Str01b]. The right translation is used in (2.22) and is written compact

$$\dot{H}_i^j = \tilde{T}_i^{j,j} H_i^j \tag{2.23}$$

The left translation leads to

$$\dot{H}_i^j = H_i^j \tilde{T}_i^{i,j} \tag{2.24}$$

We call $\tilde{T} \in T_e SE(3)$ a twist and the $\mathfrak{se}(3)$ the space of twists. It can already be seen from the upper equations that different representations exist:

- $T_i^{k,j}$ is the twist of Ψ_i with respect to Ψ_j expressed in the frame Ψ_k
- $T_i^j = T_i^{j,j}$ is the twist of Ψ_i with respect to Ψ_j expressed naturally in Ψ_j

The 4×4 matrix $\tilde{T}$ can be decomposed and there exists also a vector representation $T \in \mathbb{R}^6$

$$\tilde{T} = \begin{pmatrix} \tilde{\omega} & v \\ 0 & 0 \end{pmatrix}, \; T = \begin{pmatrix} \omega \\ v \end{pmatrix}$$

Wherein v is the velocity and ω is the angular velocity. $\tilde{\omega}$ is the skew-symmetric representation of the vector ω

$$\omega = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} \Rightarrow \tilde{\omega} = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & \omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} \tag{2.25}$$

Changes of coordinates for twists are of the form

$$\tilde{T}_i^{j,j} = \tilde{T}_i^j = H_i^j \tilde{T}_i^{i,j} H_j^i$$

or for the vector representation

$$T_i^j = Ad_{H_i^j} T_i^{i,j} \ , \ Ad_{H_i^j} = \begin{pmatrix} R_i^j & 0 \\ \tilde{p}_i^j R_i^j & R_i^j \end{pmatrix}$$

The dual vector space of $\mathfrak{se}(3)$ is the space of linear operations from $\mathfrak{se}(3)$ to $\mathbb{R}$. It is denoted by $\mathfrak{se}^*(3)$ and represents the space of wrenches W.
Wrenches decompose to moments m and forces f.

$$W = (m \ \ f) \ , \ \tilde{W} = \begin{pmatrix} \tilde{f} & m \\ 0 & 0 \end{pmatrix}$$

Again $\tilde{W} \in \mathbb{R}^{4\times4}$ is a matrix while $W \in \mathbb{R}^6$ is the row vector representation.
The change of coordinates for twists is similar to the case of twists:

$$(W^i)^T = Ad^T_{H_i^j}(W^j)^T$$

Here the mapping is in the opposite direction, from Ψ_j to Ψ_i, what is a consequence of the fact that wrenches are duals to twists. Again there are different representations of wrenches:

- $W_i^{k,j}$ is the wrench applied to a spring connecting Ψ_i to Ψ_j on the side of Ψ_i expressed in the coordinate frame Ψ_k.
- W_i^k is the wrench applied to a body attached to Ψ_i expressed in the coordinate frame Ψ_k.

Connecting a body and spring in the point Ψ_i and applying the principle of action and reaction we get $W_i^{k,j} = -W_i^k$. Due to the nodicity of a spring we have $W_i^{k,j} = -W_j^{k,i}$

Recalling Subsection 2.1.1 the power flowing through a port is defined by the dual product of an effort and flow pair, which are from a vector space and its dual respectively. On a vector space level the power port $\mathcal{P}$ is defined by the Cartesian product of the Lie algebra $\mathfrak{se}(3)$ and its dual $\mathfrak{se}^*(3)$: $\mathcal{P} = \mathfrak{se}(3) \times \mathfrak{se}^*(3)$. The mechanical power exchanged at the port is simply $P = WT$.

2.2.2 Dynamics of physical components

Springs

A *spring* is the ideal, lossless element storing potential energy. It is connected to two bodies and is defined by an energy function. The energy is a function of the relative displacement of the attached bodies. Consider a spring between the two bodies B_i and B_j, with coordinate frames Ψ_i and Ψ_j fixed to the respective body. The stored potential energy is positive definite function of the form [SMA99]

$$V_{i,j} : SE(3) \to \mathbb{R};\ H_i^j \mapsto V_{i,j}(H_i^j) \tag{2.26}$$

Recalling the general equations of a pHs and the assignment of the variables from Subsection 2.1 we already have the structure of the pHs equations for a spring. The state variable is the relative displacement H_i^j, the effort variable is the wrench $W_i^{j,j}$ and the flow variable is the twist T_i^j

$$\begin{aligned} \dot{H}_i^j &= T_i^j H_i^j \\ W_i^{j,j} &= \frac{\partial V_{i,j}}{\partial H_i^j}(H_i^j)^T \end{aligned} \tag{2.27}$$

Note that $V_{i,j}$ is an energetic minimum when H_i^j is the identity matrix I_4. An energetic minimum is physically necessary, otherwise infinite energy would be extractable. With $H_i^j = I_4$ the frames Ψ_i and Ψ_j coincide. It is possible to define springs with non-zero rest-length by introducing coordinate frames Ψ_{ic} and Ψ_{jc} rigidly attached to Ψ_i and Ψ_j respectively. The spring is now between the new frames, thus the energetic minimum is $H_{ic}^{jc} = I_4$. The displacements H_i^{ic}, H_j^{jc} are the resulting rest-lengths. Frames Ψ_{ic} and Ψ_{jc} can be chosen to represent the *center of stiffness*, where translation and rotation are maximally decoupled [SMA99].

Variable rest-length springs

Varying the rest-lengths of springs in manipulator-object interaction allows to perform a grasp and specify grasping forces. The rest-length influences the energy configuration, to allow for controlled changes of rest-length an additional power port is introduced. The chosen hinge points Ψ_b, Ψ_j of the spring define an axis, known as the principal axes of stiffness. Changes of the rest-length leave this axis unaffected, i.e. the displacement is in-line. With reference to Fig. 2.2, the change of rest-length is a change of relative displacement of Ψ_b and Ψ_i: H_i^b. Towards a PHS

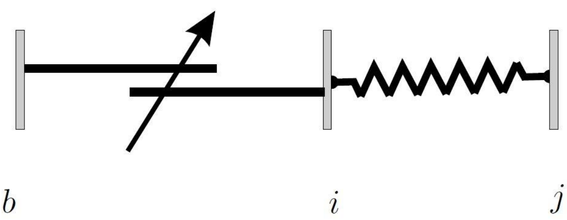

Figure 2.2: Variable rest-length spring [SMA99]

representation we need to identify the components contributing to the deformation twist of the spring T_i^j: the displacement twist of the bodies attached to the spring T_b^j and the twist resulting from a commanded change of the rest-length T_i^b [SD01].

$$T_i^j = T_b^j + Ad_{H_b^j} T_i^b \tag{2.28}$$

Since no additional energy storages were introduced, the Hamiltonian function is equal to the case of the simple spring. We can write the two-port Hamiltonian system of a variable length spring as

$$\begin{aligned} \dot{H}_i^j &= \left(\begin{pmatrix} 1 & Ad_{H_b^j} \end{pmatrix} \begin{pmatrix} T_b^j \\ T_i^b \end{pmatrix} \right) H_i^j \\ \begin{pmatrix} W_b^{j,j} \\ W_i^{b,b} \end{pmatrix} &= \left(\begin{pmatrix} 1 \\ Ad_{H_b^j}^T \end{pmatrix} \frac{\partial V_{i,j}}{\partial H_i^j} \right) (H_i^j)^T \end{aligned} \tag{2.29}$$

Inertias

Inertias are special since they in general store two types of energy: kinetic and potential energy due to gravitation. At first we exclude the gravitational terms and focus on motion. Kinetic energy is a function of the relative motion w.r.t. an inertial reference. When expressing motion in non-inertial or accelerated reference frames, fictitious forces such as the *Coriolis* or the *centrifugal* force need to be considered. Let us start from *Newton's* second law of dynamics, the time derivative of a body's momentum is equal to the applied wrench.

$$\dot{P}_b^0 = W_b^0 \tag{2.30}$$

The momentum of the inertia b and the wrench acting on it are both expressed in the inertial reference frame Ψ_0. Let us consider the non-inertial frame Ψ_b, fixed to the inertia. We start by changing coordinates, clearly we have $W_b^0 = Ad_{H_0^b}^T W_b^b$. It is detailed in [Str01b] that $P_0^b \in \mathfrak{se}^*(3)$ and thus the same transformation as for wrenches applies $P_b^0 = Ad_{H_0^b}^T P_b^b$. Expressing the *Newton's* second law in the non-inertial frame Ψ_b we have

$$\frac{d}{dt}(Ad_{H_0^b}^T P_b^b) = Ad_{H_0^b}^T W_b^b \tag{2.31}$$

The evolution of the accelerated body frame Ψ_b w.r.t to the inertial frame is time dependent. The time derivative of the transformation is $\frac{d}{dt} Ad_{H_0^b}^T = -Ad_{H_0^b}^T ad_{T_b^{b,0}}^T$. With the *adjoint* representation (see for example [Str01a])

$$ad_{T_b^{b,0}}^T = \begin{pmatrix} -\tilde{\omega}_b^{b,0} & -\tilde{v}_b^{b,0} \\ 0 & -\tilde{\omega}_b^{b,0} \end{pmatrix} \tag{2.32}$$

The second law of dynamics expressed in the body's frame is then

$$\begin{aligned} Ad_{H_0^b}^T \dot{P}_b^b - Ad_{H_0^b}^T ad_{T_b^{b,0}}^T P_b^b &= Ad_{H_0^b}^T W_b^b \\ \dot{P}_b^b &= ad_{T_b^{b,0}}^T P_b^b + W_b^b \end{aligned} \tag{2.33}$$

This formulation is split into its rotational and translational components, then we can exchange twist and momentum by the following operation

$$\begin{pmatrix} \dot{P}^b_{b,\omega} \\ \dot{P}^b_{b,v} \end{pmatrix} = \begin{pmatrix} -\tilde{\omega}^{b,0}_b & -\tilde{v}^{b,0}_b \\ 0 & -\tilde{\omega}^{b,0}_b \end{pmatrix} \begin{pmatrix} P^b_{b,\omega} \\ P^b_{b,v} \end{pmatrix} + W^b_b = \begin{pmatrix} \tilde{P}^b_{b,\omega} & \tilde{P}^b_{b,v} \\ \tilde{P}^b_{b,v} & 0 \end{pmatrix} \begin{pmatrix} \omega^{b,0}_b \\ v^{b,0}_b \end{pmatrix} + W^b_b \tag{2.34}$$

This clearly corresponds to the classical description of a rigid body's dynamics of the form

$$\dot{P}^b_b = M_b \dot{T}^{b,0}_b = C_b T^{b,0}_b + W^b_b \tag{2.35}$$

Here M_b describes the body's inertia and C_b accounts for Coriolis and centrifugal terms.
Towards port-Hamiltonian representation we start from the kinetic (co-)energy given by $V^*_k(T^{b,0}_b) = \frac{1}{2}(T^{b,0}_b)^T M_b T^{b,0}_b$. Formally speaking the kinetic energy is a function of the momentum. By using the relation of twist and momentum $P^b_b = M_b T^{b,0}_b$ we get the kinetic energy

$$V_k(P^b_b) = \frac{1}{2}(P^b_b)^T M_b^{-1} P^b_b \tag{2.36}$$

By differentiating the kinetic energy w.r.t to the state variable P^b_b we obtain

$$\frac{\partial V_k(P^b_b)}{\partial P^b_b} = M_b^{-1} P^b_b = T^{b,0}_b \tag{2.37}$$

Recall from Table 2.1 that the twist is the effort variable in the port-Hamiltonian representation of an inertia. The flow is the externally supplied wrenchW^b_b, thus we obtain the port-Hamiltonian representation of a rigid body, neglecting gravity

$$\begin{aligned} \dot{P}^b_b &= C_b \frac{\partial V_k(P^b_b)}{\partial P^b_b} + I_6 W^b_b \\ T^{b,0}_b &= I_6 \frac{\partial V_k(P^b_b)}{\partial P^b_b} \end{aligned} \tag{2.38}$$

In cooperative manipulation we often deal with heavy objects, it is thus inevitable to include the potential energy resulting from the gravitational field. One can think of a spring connecting the body and an inertial frame associated with the ground. This spring can be formulated as a pHs using the left translation (2.2.1)

$$\begin{aligned} \dot{H}^0_b &= H^0_b T^{b,0}_b \\ W^{b,0}_b &= (H^0_b)^T \frac{\partial V_g}{\partial H^0_b} \end{aligned} \tag{2.39}$$

Where V_g is a suitable energy function. For a combined description the potential and kinetic energy add up: $V_{kg} = V_k + V_g$. Since there are two types of energy stored by *one* body, the twists in both energy systems are equal. The wrenches on the body add up

$$W^b_{kg} = W^b_b + C_b \frac{\partial V_{kg}}{\partial P^b_b} - (H^0_b)^T \frac{\partial V_{kg}}{\partial H^0_b}$$

Note that the negative sign in the upper equation comes from $W_b^b = -W_b^{b,0}$.
With this knowledge we can write the combined PHS representation

$$\begin{aligned}
\begin{pmatrix} \dot{H}_b^0 \\ \dot{P}_b^b \end{pmatrix} &= \begin{pmatrix} 0 & H_b^0 \\ -(H_b^0)^T & C_b \end{pmatrix} \begin{pmatrix} \frac{\partial V_{kg}}{\partial H_b^0} \\ \frac{\partial V_{kg}}{\partial P_b^b} \end{pmatrix} + \begin{pmatrix} 0 \\ I_6 \end{pmatrix} W_b^b \\
T_b^{b,0} &= \begin{pmatrix} 0 & I_6 \end{pmatrix} \begin{pmatrix} \frac{\partial V_{kg}}{\partial H_b^0} \\ \frac{\partial V_{kg}}{\partial P_b^b} \end{pmatrix}
\end{aligned} \tag{2.40}$$

Dampers

Dampers do not have a state since they do not store energy, they only dissipate it. Note that energy is not "destroyed" in the dampers but transformed into thermal energy. This can be modelled with a thermal port connected to the environment, since we are only interested in free mechanical energy this port is omitted. The easiest way to achieve damping is a linear resistive element R, such that the wrench is directly proportional to twist. Consider for example a body's motion with respect to the inertial frame

$$W_b^b = RT_b^{b,0} \tag{2.41}$$

Or a damper in parallel with spring

$$W_i^{j,j} = RT_i^j \tag{2.42}$$

The dissipated co-energy is $E_d = \frac{1}{2}T^T RT$.

2.3 Imposing constraints

The interconnection of a spring and an inertia is the the ideal pair in terms of input-output causality. The spring expects a twist-input and outputs a wrench. The inertia has a wrench-input and outputs a twist. It can be seen from eq. (2.40) that the interconnection of inertia and spring gives a set of ordinary differential equations (ODE). Many mechanical systems cannot be modelled by an interconnection of springs and masses. For example the rigid coupling of two masses already restricts their possible motion to the degrees of freedom of a single body. The attempt to move the bodies individually results in *internal* forces. The motion-limiting conditions are called kinematic *constraints* and are expressed in the form

$$A^T(q)\dot{q} = 0 \tag{2.43}$$

We call $A(q)$ the *constraint* matrix. The associated internal forces are given by $A(q)\lambda$, the *Lagrange* multipliers Λ are uniquely defined if the constraints are satisfied. In this case the constraint forces do not influence the energy of the system since

$\lambda^T(A^T(q)\dot{q}) = 0$. Consequently the port-Hamiltonian system becomes a mixed set of differential and algebraic equations (DAE).

$$\begin{aligned}\begin{pmatrix}\dot{q}\\ \dot{p}\end{pmatrix} &= \begin{pmatrix}0 & I\\ -I & 0\end{pmatrix}\begin{pmatrix}\frac{\partial V}{\partial q}\\ \frac{\partial V}{\partial p}\end{pmatrix} + \begin{pmatrix}0 & 0\\ A(q) & g(q)\end{pmatrix}\begin{pmatrix}\lambda\\ f\end{pmatrix}\\ \begin{pmatrix}0\\ e\end{pmatrix} &= \begin{pmatrix}0 & A^T(q)\\ 0 & g^T(q)\end{pmatrix}\begin{pmatrix}\frac{\partial V}{\partial p}\\ \frac{\partial V}{\partial p}\end{pmatrix}\end{aligned} \tag{2.44}$$

The system is no longer described in the input-output form, but in an implicit form. Several approaches to solve the algebraic equations and restore the desired input-output form exist. We introduce two methods leading to different results, first in a simple symbolic representation and then apply them two the 6-DoF space.
The first method (described e.g. in [DS09]) uses the time-derivation of the constraints.

$$0 = \frac{d}{dt}\left(A^T(q)\frac{\partial V}{\partial p}\right) \tag{2.45}$$

We use the property $\frac{\partial V}{\partial p} = M^{-1}p$ for an energy function of the form $V = \frac{1}{2}p^T M^{-1}p + V_q$. Since $q(t)$ is time-dependent *indirect dependencies* arise in $A(q), p(q)$ when calculating the total time-derivative

$$\begin{aligned}0 &= \frac{d}{dt}(A^T M^{-1}p) = \frac{\partial(A^T M^{-1}p)}{\partial q}\dot{q} + A^T M^{-1}\dot{p} =\\ &= \frac{\partial(A^T M^{-1}p)}{\partial q}M^{-1}p + A^T M^{-1}\left(-\frac{\partial V}{\partial q} + A(q)\lambda + g(q)f\right)\end{aligned} \tag{2.46}$$

Solving this equation for λ gives

$$\lambda = (A^T M^{-1}A)^{-1}\left(-\frac{\partial(A^T M^{-1}p)}{\partial q}M^{-1}p + \frac{\partial V}{\partial q} - gf\right) \tag{2.47}$$

This is an analytic expression for the constrained forces, λ is re-inserted into eq. (2.44) and we obtain a set of ODEs in input-output form. Note that the forces λ keep the velocities $A(q)\dot{q}$, i.e. if the constraint $A(q)\dot{q} = 0$ was satisfied in the beginning it will remain satisfied.
Another approach is to find new coordinates for the momentum that limit the *phase space* (q, p) to motions fulfilling the constraints. Therefore we introduce the transformation

$$p = S(q)\bar{p} \tag{2.48}$$

The definition of $S(q)$ is directly given by the constraint

$$0 = A^T(q)M^{-1}p = A^T(q)M^{-1}S(q)\bar{p} \tag{2.49}$$

S(q) is thus the *kernel* of $A^T(q)M^{-1}$. The transformation is energy-conservative i.e.

$$\bar{V}(q,\bar{p}) := \frac{1}{2}\bar{p}^T\bar{M}^{-1}\bar{p} + V_q = \frac{1}{2}p^T M^{-1}p + V_q = V(q,p) \tag{2.50}$$

Thus the constrained inertia matrix is $\bar{M} = (S^T M^{-1} S)^{-1}$. We start re-writing eq. (2.44) using $p = S(q)\bar{p}$ and obtain

$$\begin{aligned} \dot{q} &= \frac{\partial V}{\partial p} = M^{-1}p = M^{-1}S\bar{p} = M^{-1}S\bar{M}\bar{M}^{-1}\bar{p} = M^{-1}S\bar{M}\frac{\partial \bar{V}}{\partial \bar{p}} \\ \dot{p} &= \frac{d}{dt}(S(q)\bar{p}) = \frac{\partial (S\bar{p})}{\partial q}\dot{q} + S\dot{\bar{p}} = -\frac{\partial V}{\partial q} + A\lambda + gf \end{aligned} \tag{2.51}$$

Towards a standard port-Hamiltonian representation we are seeking to express the last equation in the form $\dot{\bar{p}} = ...$ Therefore the equation is pre-multiplied with $\bar{M}S^T M^{-1}$

$$\bar{M}S^T M^{-1}\frac{\partial (S\bar{p})}{\partial q}\dot{q} + \bar{M}\underbrace{S^T M^{-1} S}_{\bar{M}^{-1}}\dot{\bar{p}} = \bar{M}S^T M^{-1}\left(-\frac{\partial V}{\partial q} + A\lambda + gf\right) \tag{2.52}$$

For a symmetric M we have $(S^T M^{-1} A)^T = A^T(q)M^{-1}S(q) = 0$. Since $S(q)$ is dependent on q we need to consider indirect dependencies when differentiating

$$\frac{\partial \bar{V}}{\partial q} = \frac{\partial V}{\partial q} + \frac{\partial^T (S(q)\bar{p})}{\partial q}\dot{q} = \frac{\partial V}{\partial q} + \frac{\partial^T (S(q)\bar{p})}{\partial q}M^{-1}S\bar{M}\frac{\partial \bar{V}}{\partial \bar{p}} \tag{2.53}$$

Inserting this into eq. (2.52) and re-arranging we desired port-Hamiltonian representation

$$\begin{aligned} \begin{pmatrix} \dot{q} \\ \dot{\bar{p}} \end{pmatrix} &= \begin{pmatrix} 0 & M^{-1}S\bar{M} \\ \bar{M}S^T M^{-1} & \bar{M}S^T M^{-1}\left(\frac{\partial^T (S(q)\bar{p})}{\partial q} - \frac{\partial (S(q)\bar{p})}{\partial q}\right)M^{-1}S\bar{M} \end{pmatrix} \begin{pmatrix} \frac{\partial \bar{V}}{\partial q} \\ \frac{\partial \bar{V}}{\partial \bar{p}} \end{pmatrix} + \\ &+ \begin{pmatrix} 0 \\ \bar{M}S^T M^{-1} \end{pmatrix} f \end{aligned} \tag{2.54}$$

Conclusively we have obtained two different representations of constrained systems in input-output form. The first one allows explicitly to calculate the constraint forces necessary to comply with the constraints. The second one restricts the possible motion and results in a system of reduced *degrees of freedom*. We see in the remainder that the first can be exploited for control issues. The latter gives a system model, useful when describing the response to an external wrench acting on the common object.

Constraints for 6D-motion

When it comes to constraints in six dimensions of freedom it is necessary to revisit the concept of twists used throughout this thesis. Recall the definition of a twist from eq. (2.23) being

$$\begin{aligned} T_i^j = \dot{H}_i^j H_j^i &= \begin{pmatrix} \dot{R}_i^j & \dot{p}_i^j \\ 0 & 0 \end{pmatrix} \begin{pmatrix} R_j^i & p_j^i \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \dot{R}_i^j & \dot{p}_i^j \\ 0 & 0 \end{pmatrix} \begin{pmatrix} (R_i^j)^T & -(R_i^j)^T p_i^j \\ 0 & 1 \end{pmatrix} = \\ &= \begin{pmatrix} \dot{R}_i^j (R_i^j)^T & -\dot{R}_i^j (R_i^j)^T p_i^j + \dot{p}_i^j \\ 0 & 0 \end{pmatrix} =: \begin{pmatrix} \tilde{\omega}_i^j & v_i^j \\ 0 & 0 \end{pmatrix} \end{aligned} \tag{2.55}$$

It is clear from this equation that the linear velocity part v_i^j is not the velocity of the frame Ψ_i with respect to Ψ_j, identified by $\dot{p}_i^j$. This twist representation is described by the *screw theory* (see e.g. [WS08]). It can be visualized by the angular velocity around an axis and the linear velocity along this axis. Consider rigidly connected two bodies, associated with the frames Ψ_b and Ψ_i and a distance between them $p_i^b = p_i^0 - p_b^0$. Clearly, in the setting of cooperative manipulation, one can think of an object b and the i-th manipulator attached to it. Now let the body b rotate with the angular velocity ω_b^0. Being rigidly attached the body i rotates in the same manner, $\omega_i^0 = \omega_b^0$. The translation of body i is expressed dependent on body b by

$$\begin{aligned} \dot{p}_i^0 &= \dot{p}_b^0 + \omega_b^0 \times p_i^b = \dot{p}_b^0 + \omega_b^0 \times (p_i^0 - p_b^0) \\ \dot{p}_i^0 - \omega_b^0 \times p_i^0 &= \dot{p}_b^0 - \omega_b^0 \times p_b^0 \\ v_i^0 &= v_b^0 \end{aligned} \tag{2.56}$$

This immediately leads to $T_i^0 = T_b^0$. For a system if $i = 1...N$ manipulators we can thus write the constraining condition

$$0 = A^T T = \begin{pmatrix} -I_3 & 0_3 & I_3 & 0_3 & & & \\ 0_3 & -I_3 & 0_3 & I_3 & & & \\ \vdots & \vdots & & & \ddots & & \\ -I_3 & 0_3 & & & & I_3 & 0_3 \\ 0_3 & -I_3 & & & & 0_3 & I_3 \end{pmatrix} \begin{pmatrix} T_b^0 \\ T_1^0 \\ \vdots \\ T_N^0 \end{pmatrix} \tag{2.57}$$

We start applying the first method presented int the previous subsection by differentiating the constraints. The constraint matrix A is not time dependent, eq. (2.45) simplifies to

$$0 = \frac{d}{dt} A^T T = A^T \dot{T} = A^T M^{-1} \dot{P} \tag{2.58}$$

First of all we look at the twist's time-derivative and obtain an equivalent formulation of constraint dynamics as for a classic approach e.g. [EH16]. Consider the time-derivative of the constraint for the rigidly connected bodies b and i.

$$\begin{aligned} 0 = \dot{T}_i^0 - \dot{T}_b^0 &= \begin{pmatrix} \dot{\omega}_i^0 - \dot{\omega}_b^0 \\ \dot{v}_i^0 - \dot{v}_b^0 \end{pmatrix} = \begin{pmatrix} \dot{\omega}_i^0 - \dot{\omega}_b^0 \\ \frac{d}{dt}(\dot{p}_i^0 - \omega_b^0 \times p_i^0) - \frac{d}{dt}(\dot{p}_b^0 - \omega_b^0 \times p_b^0) \end{pmatrix} = \\ &= \begin{pmatrix} \dot{\omega}_i^0 - \dot{\omega}_b^0 \\ \ddot{p}_i^0 - \ddot{p}_b^0 - \dot{\omega}_b^0 \times p_i^b - \omega_b^0 \times (\omega_b^0 \times p_i^b) \end{pmatrix} \end{aligned} \tag{2.59}$$

The chosen twist representation fulfils the second-order dynamic requirements including centripetal terms.

Towards solving the set of DAEs, we insert the port-Hamiltonian representations into the constraint equation

$$0 = A^T M^{-1} \dot{P} = A^T M^{-1}(W + CT + A\lambda) \tag{2.60}$$

and solve for λ

$$\lambda = -(A^T M^{-1} A)^{-1} A^T M^{-1}(W + CT) \tag{2.61}$$

Consider again the example of two rigidly connected bodies b and i. Let us first examine the part $A^T M^{-1} CT$, using eq. (2.34) and assuming that the bodies share the orientation $R_0^b = R_0^i$, we obtain

$$\begin{aligned}
A^T M^{-1} \begin{pmatrix} C_b T_b^{b,0} \\ C_i T_i^{i,0} \end{pmatrix} &= \begin{pmatrix} 0 \\ \tilde{\omega}_b^{b,0} v_b^{b,0} - \tilde{\omega}_i^{i,0} v_i^{i,0} \end{pmatrix} = \\
&= \begin{pmatrix} 0 \\ \tilde{\omega}_b^{b,0} \tilde{p}_b^0 R_0^b \omega_b^0 + \tilde{\omega}_b^{b,0} R_0^b v_b^0 - \tilde{\omega}_b^{b,0} \tilde{p}_i^0 R_0^b \omega_b^0 - \tilde{\omega}_b^{b,0} R_0^b v_i^0 \end{pmatrix} = \\
&= \begin{pmatrix} 0 \\ \tilde{\omega}_b^{b,0} (\tilde{p}_b^0 - \tilde{p}_i^0) \omega_b^{b,0} \end{pmatrix} = \begin{pmatrix} 0 \\ \tilde{\omega}_b^{b,0} \tilde{p}_b^i \omega_b^{b,0} \end{pmatrix}
\end{aligned} \tag{2.62}$$

Once again we have obtained a term for the centripetal acceleration, the resulting λ is

$$\lambda = \left[A^T M^{-1} A)^{-1} (M_b^{-1} W_b^b - M_i^{-1} W_i^i + \begin{pmatrix} 0 \\ \omega_b^{b,0} \times (\omega_b^{b,0} \times p_b^i) \end{pmatrix} \right] \tag{2.63}$$

This example shows how kinematic constraints can be solved by calculating the constraint forces. The results are equivalent to approaches based on the *Gauss' principle of least constraint* ([EH16]) or *Euler-Lagrange* representations ([**?**]).
The other fore-mentioned approach of restricting the possible motion starts with finding a matrix S fulfilling $0 = A^T M^{-1} S \bar{P}$ for every $\bar{P}$. Thus we require $0 = A^T M^{-1} S$. Consider the example of two rigidly connected bodies b and i, we can write the constraint equation

$$0 = A^T \begin{pmatrix} T_b^0 \\ T_i^0 \end{pmatrix} = A^T \begin{pmatrix} Ad_{H_b^0} & 0 \\ 0 & Ad_{H_i^0} \end{pmatrix} \begin{pmatrix} M_b^{-1} & 0 \\ 0 & M_i^{-1} \end{pmatrix} \begin{pmatrix} P_b^b \\ P_i^i \end{pmatrix} \tag{2.64}$$

Now we introduce the transformation $P = S\bar{P}$ In an object-centred cooperative manipualtion case $\bar{P}$ clearly corresponds to the twist of the common object. Starting from $T_i^0 = T_b^0$ towards $T_i^{i,0} = Ad_{H_b^i} T_b^{b,0}$ we find

$$S = \begin{pmatrix} I_3 & 0 \\ 0 & I_3 \\ j_i j_b^{-1} & 0 \\ m_i \tilde{p}_b^i j_b^{-1} & m_i m_b^{-1} \end{pmatrix} \tag{2.65}$$

Constraints on rigid bodies

Consider a group of rigidly connected bodies, in the center of the group the *virtual object* associated with the coordinate frame Ψ_v. The bodies surrounding it share the same orientation and the position of the i-th body is

$$p_i^0 = p_v^0 + R_v^0 p_i^v$$

By differentiating the *geometric* constraints we obtain the *kinematic* constraints

$$\dot{p}_i^0 = \dot{p}_v^0 + \omega_v \times p_i^v$$

$$\omega_i = \omega_v$$

We can reformulate these equations in matrix vector notation and $i = 1...N$

$$\underbrace{\begin{pmatrix} -I_3 & 0_3 & I_3 & 0_3 & & & \\ \tilde{p_1^v} & -I_3 & 0_3 & I_3 & \cdots & 0_3 & 0_3 \\ \vdots & \vdots & & & \ddots & & \\ -I_3 & 0_3 & & & & I_3 & I_3 \\ \tilde{p_N^v} & -I_3 & & & & 0_3 & I_3 \end{pmatrix}}_{A^T(q)} \underbrace{\begin{pmatrix} \omega_v \\ \dot{p}_v^0 \\ \omega_1 \\ \dot{p}_1^0 \\ \vdots \\ \omega_N \\ \dot{p}_N^0 \end{pmatrix}}_{\dot{q}} = 0 \tag{2.66}$$

In order to eliminate the constraint forces from equation (2.44) we require the *kernel* of $A^T(q)$, given by

$$S(q) = ker\, A^T = \begin{pmatrix} I_3 & 0_3 \\ 0_3 & I_3 \\ I_3 & 0_3 \\ -\tilde{p}_1^v & I_3 \\ \vdots & \vdots \\ I_3 & 0_3 \\ -\tilde{p}_N^v & I_3 \end{pmatrix}$$

This allows to employ the coordinate transformation (**??**) and obtain the constrained momentum. Choosing a representation of the form $\bar{p} = \bar{M}\dot{q}$ is advantageous when deriving the transformed *Hamiltonian*

$$\bar{p} = \begin{pmatrix} j_v + \sum_{i=1}^N [j_i - m_i(\tilde{p}_i^v)^2] & -\sum_{i=1}^N m_i \tilde{p}_i^v \\ \sum_{i=1}^N m_i \tilde{p}_i^v & m_v + \sum_{i=1}^N m_i \end{pmatrix} \begin{pmatrix} \omega_v \\ \dot{p}_v^0 \end{pmatrix}$$

Note that $\bar{M}$ is symmetric, this is due to the skew-symmetry of $\tilde{p}_i^v$. Thus the *Hamiltonian* is obtained as usual by $\bar{V} = \frac{1}{2}\bar{p}^T \bar{M}^{-1} \bar{p}$. One can compute that $\frac{\partial S_i}{\partial q} = 0\ \forall i = 1...N$, thus the *Lie bracket* $[S_i, S_j]$ equals zero.
The external wrenches acting on the free bodies sum up for the constrained system, respecting the geometric requirements we obtain

$$\bar{g} = \begin{pmatrix} I_3 & 0_3 & I_3 & \tilde{p}_1^v & \cdots & I_3 & \tilde{p}_N^v \\ 0_3 & I_3 & 0_3 & I_3 & \cdots & 0_3 & I_3 \end{pmatrix}$$

Towards the port-Hamiltonian representation of the constrained system we seek to replace the term $\frac{\partial \bar{V}}{\partial q}$ by $\frac{\partial \bar{V}}{\partial \bar{p}}$. The kinetic co-energy of the system is $\bar{V}^* = \frac{1}{2}\dot{q}^T \bar{M} \dot{q}$.

2.4 Spring-mass-damper systems

Recall the motivating example from Section 2.1 of a simple spring-mass system. We can add a damper d to the pHs representation and rewrite eq. (2.5)

$$\begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & -d \end{pmatrix} \begin{pmatrix} \frac{\partial V}{\partial q}(q,p) \\ \frac{\partial V}{\partial p}(q,p) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} F_e \tag{2.67}$$

Clearly $\dot{p} = -\frac{\partial V}{\partial q}(q,p) - d\frac{\partial V}{\partial p}(q,p) + F_e$ equals the mechanical impedance relation $m\ddot{x} + d\dot{x} + kx = F_e$. Thus we have a port-Hamiltonian representation of the impedance control scheme. This is the basis for the control architecture presented in the next chapter. Therefore the mass-spring-damper system, given as a one-dimensional example, is formulated in the $SE(3)$. We start from eqs. (2.38,2.39) and add another spring to the body associated with Ψ_b. This spring connects to a desired object position assigned to Ψ_v. Its pHs representation is given by

$$\begin{aligned} \dot{H}_b^v &= H_b^v T_b^{b,v} \\ W_b^{b,v} &= (H_b^v)^T \frac{\partial V_s}{\partial H_b^v} \end{aligned} \tag{2.68}$$

The spring's deformation twist is decomposed by $T_b^{b,v} = T_b^{b,0} - T_v^{b,0}$. The damping along this spring is $W_b^b = D_b T_b^{b,v}$. Body, spring and damper move uniformly with the twist $T_b^{b,0}$ and the wrenches add up. Combing all components we arrive at

$$\begin{aligned} \begin{pmatrix} \dot{H}_b^0 \\ \dot{H}_b^v \\ \dot{P}_b^b \end{pmatrix} &= \begin{pmatrix} 0 & 0 & H_b^0 \\ 0 & 0 & H_b^v \\ -(H_b^0)^T & -(H_b^v)^T & C_b - D_b \end{pmatrix} \begin{pmatrix} \frac{\partial V_{kgs}}{\partial H_b^0} \\ \frac{\partial V_{kgs}}{\partial H_b^v} \\ \frac{\partial V_{kgs}}{\partial P_b^b} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -H_b^v Ad_{H_0^b} & 0 \\ D_b Ad_{H_0^b} & I_6 \end{pmatrix} \begin{pmatrix} T_v^0 \\ W_b^b \end{pmatrix} \\ \begin{pmatrix} W_v^{0,0} \\ T_b^{b,0} \end{pmatrix} &= \begin{pmatrix} 0 & -Ad_{H_0^b}^T (H_b^v)^T & Ad_{H_0^b}^T D_b \\ 0 & 0 & I_6 \end{pmatrix} \begin{pmatrix} \frac{\partial V_{kgs}}{\partial H_b^0} \\ \frac{\partial V_{kgs}}{\partial H_b^v} \\ \frac{\partial V_{kgs}}{\partial P_b^b} \end{pmatrix} \end{aligned} \tag{2.69}$$

This clearly accounts for an *external* impedance relation, used to establish compliant behaviour between (virtual) object and environment. Analogously we can define impedance relations between manipulators and virtual object. Here we consider the manipulator masses to be gravity pre-compensated and omit the spring connecting to the ground. Rest-length springs are used to allow for formation changes. The i-th manipulator inertia is given by

$$\begin{aligned} \dot{P}_i^i &= C_i \frac{\partial V_{k(i)}(P_i^i)}{\partial P_i^i} + I_6 W_i^i \\ T_i^{i,0} &= I_6 \frac{\partial V_{k(i)}(P_i^i)}{\partial P_i^i} \end{aligned} \tag{2.70}$$

A variable rest-length spring as defined in eq. 2.29) connects to the body associated with Ψ_i. The rest-length is $H^b_{b(i)}$. The spring's twist decomposes as follows

$$T^i_{b(i)} = T^i_b + T^{i,b}_{b(i)} = Ad_{H^i_b} T^{b,0}_b - T^{i,0}_i + Ad_{H^i_b} T^b_{b(i)} \tag{2.71}$$

Thus the variable rest-length spring is given by

$$\begin{aligned}
\dot{H}^i_{b(i)} &= H^i_{b(i)} \begin{pmatrix} Ad_{H^{b(i)}_b} & -Ad_{H^{b(i)}_i} & Ad_{H^{b(i)}_b} \end{pmatrix} \begin{pmatrix} T^{b,0}_b \\ T^{i,0}_i \\ T^b_{b(i)} \end{pmatrix} \\
\begin{pmatrix} W^{b,0}_b \\ W^{i,0}_i \\ W^{b,b}_{b(i)} \end{pmatrix} &= \begin{pmatrix} Ad^T_{H^{b(i)}_b} \\ -Ad^T_{H^{b(i)}_i} \\ Ad^T_{H^{b(i)}_b} \end{pmatrix} (H^i_{b(i)})^T \frac{\partial V_{s(i)}}{\partial H^i_{b(i)}}
\end{aligned} \tag{2.72}$$

Considering damping there are two options, the relative twist of the spring tips $T^i_{b(i)}$ or the whole rest-length spring twist T^i_b can be subject to damping. In order to obtain a simpler representation we choose the latter being

$$W^i_i = D_i T^{i,b}_i = D_i T^{i,0}_i - D_i Ad_{H^i_b} T^{b,0}_b \tag{2.73}$$

This means that changes of rest-length are also subject to damping. We can combine spring, inertia and damper, the twist $T^{i,0}_i$ is the common quantity.

$$\begin{aligned}
\begin{pmatrix} \dot{H}^i_{b(i)} \\ \dot{P}^i_i \end{pmatrix} &= \begin{pmatrix} 0 & H^i_{b(i)} Ad_{H^{b(i)}_i} \\ -Ad^T_{H^{b(i)}_i} (H^i_{b(i)})^T & C_i - D_i \end{pmatrix} \begin{pmatrix} \frac{\partial V_{ks(i)}}{\partial H^i_{b(i)}} \\ \frac{\partial V_{ks(i)}}{\partial P^i_i} \end{pmatrix} + \\
&+ \begin{pmatrix} H^i_{b(i)} Ad_{H^{b(i)}_b} & H^i_{b(i)} Ad_{H^{b(i)}_b} & 0 \\ D_i Ad_{H^i_b} & 0 & I_6 \end{pmatrix} \begin{pmatrix} T^{b,0}_b \\ T^b_{b(i)} \\ W^i_i \end{pmatrix} \\
\begin{pmatrix} W^{b,0}_b \\ W^{b,b}_{b(i)} \\ T^{i,0}_i \end{pmatrix} &= \begin{pmatrix} Ad^T_{H^{b(i)}_b} (H^i_{b(i)})^T & Ad^T_{H^i_b} D_i \\ Ad^T_{H^{b(i)}_b} (H^i_{b(i)})^T & 0 \\ 0 & I_6 \end{pmatrix} \begin{pmatrix} \frac{\partial V_{ks(i)}}{\partial H^i_{b(i)}} \\ \frac{\partial V_{ks(i)}}{\partial P^i_i} \end{pmatrix}
\end{aligned} \tag{2.74}$$

Chapter 3

port-Hamiltonian based control

Recall the power balance already given by eq. (2.14), in integrated form we obtain the energy balance of a pHs

$$\underbrace{H(x(t)) - H(x(0))}_{stored\,energy} = \underbrace{\int_0^t e_P^T f_P dt}_{ext.\,supplied} + \underbrace{\int_0^t e_R^T f_R dt}_{dissipated} \tag{3.1}$$

The dissipated co-energy is negative since $e_R^t f_R \leq 0$. Clearly the given energy balance corresponds to a *passive* system. The Hamiltonian energy functions of the systems presented in Chapter 2 are *candidate Lyapunov* functions. The interconnection of pH/passive systems is still a pH/passive system. By interconnection of pH-controller and passive plant we obtain a passive controlled-robotic system. This type of systems has an important advantage when it comes to interaction with humans and/or unknown environments. Usually whose dynamics are either uncertain or too complicated to be modelled. Passive systems are stable with any system, regardless its structure or complexity, that can provide only a bounded amount of energy (see [Str15] for details). In this chapter we propose a passive model-based controller for a robot-team manipulating a common object. We employ *collocated control*, i.e. for the controller we use only the information contained in the effort-flow pair connecting the controller to the robotic system. There are no additional observers (e.g. tracking of the manipulated object).

3.1 Control architecture

Starting from the mechanical impedance equations derived in subsection 2.4, a controller based on the structure of the cooperative manipulation set-up is designed. The starting point is the impedance equation (2.69) accounting for the relation between object and reference trajectory. The inertia M_b represents the real system's common object and is thus called *virtual* object (cf. [Str01a]). Spring and damper establish a relation between desired and *actual* object twist. In this context the actual twist is the twist of the virtual object. It is important to notice that no external object tracking is employed. Analogously to a real cooperative manipulation set-up, the i-manipulators connect to the virtual object. In the controller these connections are compliant (not rigid), i.e. springs are between object and manipulators. The manipulator-object impedance equation (2.74) defines the rest-length springs establishing this connection. One hinge-point is connected to the center of the virtual object and the rest-length accounts for the extents of the object. The other hinge point is connected to the impedance relation's inertia M_i, which clearly represents the i-th manipulator. In summary a simple geometric interconnection of the impedance equations (2.69,2.74) forms the controller.

$$
\begin{pmatrix} \dot{H}_b^0 \\ \dot{H}_b^v \\ \dot{P}_b^b \\ \dot{H}_{b(1)}^1 \\ \dot{P}_1^1 \\ \vdots \\ \dot{H}_{b(N)}^N \\ \dot{P}_N^N \end{pmatrix} = \begin{pmatrix} 0 & 0 & H_b^0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & H_b^v & 0 & 0 & \cdots & 0 & 0 \\ -(H_b^0)^T & -(H_b^v)^T & C_b - D_b & Ad^T_{H_b^{b(1)}}(H_{b(1)}^1)^T & Ad^T_{H_b^1} D_1 & \cdots & Ad^T_{H_b^{b(N)}}(H_{b(N)}^N)^T & Ad^T_{H_b^N} D_N \\ 0 & 0 & H_{b(1)}^1 Ad_{H_b^{b(1)}} & 0 & H_{b(1)}^1 Ad_{H_1^{b(1)}} & & 0 & 0 \\ 0 & 0 & D_1 Ad_{H_b^1} & -Ad^T_{H_1^{b(1)}} H_{b(1)}^1 & C_1 - D_1 & & 0 & 0 \\ \vdots & \vdots & \vdots & & & \ddots & & \\ 0 & 0 & H_{b(N)}^N Ad_{H_b^{b(N)}} & 0 & 0 & & 0 & H_{b(N)}^N Ad_{H_N^{b(N)}} \\ 0 & 0 & D_N Ad_{H_b^N} & 0 & 0 & & -Ad^T_{H_N^{b(N)}} H_{b(N)}^N & C_N - D_N \end{pmatrix} \begin{pmatrix} \frac{\partial V}{\partial H_b^0} \\ \frac{\partial V}{\partial H_b^v} \\ \frac{\partial V}{\partial P_b^b} \\ \frac{\partial V}{\partial H_{b(1)}^1} \\ \frac{\partial V}{\partial P_1^1} \\ \vdots \\ \frac{\partial V}{\partial H_{b(N)}^N} \\ \frac{\partial V}{\partial P_N^N} \end{pmatrix} +
$$

$$
+ \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ -H_b^v Ad_{H_b^0} & 0 & 0 & \cdots & 0 & 0 \\ D_b Ad_{H_0^b} & 0 & 0 & \cdots & 0 & 0 \\ 0 & H_{b(1)}^1 Ad_{H_b^{b(1)}} & 0 & & 0 & 0 \\ 0 & 0 & I_6 & & 0 & 0 \\ \vdots & & & \ddots & & \\ 0 & 0 & 0 & & H_{b(N)}^N Ad_{H_b^{b(N)}} & 0 \\ 0 & 0 & 0 & & 0 & I_6 \end{pmatrix} \begin{pmatrix} T_v^0 \\ T_{b(1)}^b \\ W_1^1 \\ \vdots \\ T_{b(N)}^b \\ W_N^N \end{pmatrix} \tag{3.2}
$$

$$
\begin{pmatrix} W_v^{v,0} \\ W_{b(1)}^{b,b} \\ T_1^{1,0} \\ \vdots \\ W_{b(N)}^{b,b} \\ T_N^{N,0} \end{pmatrix} = \begin{pmatrix} 0 & -Ad^T_{H_b^0}(H_b^v)^T & Ad^T_{H_0^b} D_b & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & Ad^T_{H_b^{b(1)}}(H_{b(1)}^1)^T & 0 & & 0 & 0 \\ 0 & 0 & 0 & 0 & I_6 & & 0 & 0 \\ \vdots & \vdots & \vdots & & & \ddots & & \\ 0 & 0 & 0 & 0 & 0 & & Ad^T_{H_b^{b(N)}}(H_{b(N)}^N)^T & 0 \\ 0 & 0 & 0 & 0 & 0 & & 0 & I_6 \end{pmatrix} \begin{pmatrix} \frac{\partial V}{\partial H_b^0} \\ \frac{\partial V}{\partial H_b^v} \\ \frac{\partial V}{\partial P_b^b} \\ \frac{\partial V}{\partial H_{b(1)}^1} \\ \frac{\partial V}{\partial P_1^1} \\ \vdots \\ \frac{\partial V}{\partial H_{b(N)}^N} \\ \frac{\partial V}{\partial P_N^N} \end{pmatrix}
$$

Notably the input signal coming from the manipulator is the wrench W_i^i, thus we require force/torque sensing at the robots. Consequently the output of the controller is the twist $T_i^{i,0}$, leading to a twist-controlled robotic system. Every effort-flow pair contains not only the information encoded by its definition but also the power needed to execute the commanded action. A common concept in tele-operation is the wave transformation (see [NS04] for a survey) based on effort-flow pairs. The resulting wave variables possess *hybrid encoding*, i.e. the distinction between effort and flow is discarded. The receiving robot interprets the wave based on its needs. Wave variables only encode strength and direction of an action but do not discriminate between force and twist. This is the motivation to introduce a power-preserving transformation based on a constant factor b, that relates twists and wrenches.

$$\begin{aligned} \mathcal{W}_i^i &= bT_i^{i,0}\ ,\ i = 1...N \\ W_i^i &= b\mathcal{T}_i^{i,0} \end{aligned} \tag{3.3}$$

Where $\mathcal{W}, \mathcal{T}$ denote the wrenches and twists of the robot side. The constant b is another tunable parameter, it allows for choosing the weighting of twists and wrenches.

3.2 Human guidance

An intuitive way of guiding the robotic system is to let it follow the hand of the operator. Tracking systems for either of the hand or a device held by the hand are available. These motion capture systems usually run at lower frequencies than the robotic control loop (which is assumed time-continuous). Therefore the tracking output is not a smooth twist trajectory but a time-discrete sequence of position and orientation data. The objective of this section mainly is the derivation of the reference twist T_v^0 from the discrete tracking results. From the previous sections we know $T_v^0 = \dot{H}_v^0 H_0^v$, in a discrete representation this is

$$\begin{aligned} T_v^0(k+1) &= \dot{H}_v^0(k+1)H_0^v(k+1)(H_0^v(k))^{-1} = \\ &= \dot{H}_v^0(k+1)(H_v^0(k+1))^{-1}H_v^0(k) \end{aligned} \tag{3.4}$$

We can calculate the change of position/orientation over the sampling time interval ΔT

$$\dot{H}_v^0(k+1) = \frac{H_v^0(k+1)(H_v^0(k))^{-1}}{T_s} \tag{3.5}$$

Thus we have obtained a discrete twist representation based on discrete pose inputs. It is a discontinuous sequence updated every T_s and thus exhibits steps in its value. The energy exchanged through the port during T_s is

$$\Delta V_k^{k+1} = \int_{kT_s}^{(k+1)T_s} W_v^{v,0} dt\ T_v^0(k+1) \tag{3.6}$$

Steps of the input result in discontinuities in the energy function, since the input(flow) is the time derivative of the energy state. Consider the time-discrete twist $T_v^0(k+1)$ assigned to the continuous system input, i.e. $T_v^0(t) := T_v^0(k+1)$. Thus the input changes its value every T_s. The input $T_v^0(t)$ is an external flow f_p, contributing to the energy balance given in eq. (2.14). Since the pHs is continuous the property of energy conservation holds for all f_P. The steps in the system energy are exclusively supplied by external power and are thus not passivity violating. In contrast to there are time-discrete pHs, described in [SSVdSF05]. The main difference are time-discrete energy states, i.e. the state of the next time step is computed from the actual flow and state value. In this class of systems passivity is violated if, throughout a time-step, more energy than stored is extracted. Therefore consider the example of an elongated spring. The relaxing twist is so high that its integration (i.e. the displacement) over the time-step is higher than the initial elongation. I.e. in one time-step the spring not only fully relaxes to its equilibrium but also elongates in the other direction, this would generate energy and violates passivity. In our case assuming a continuous pHs, this cannot happen because the time-steps are infinitely small.
Using time-discrete input functions with a continuous pHs does not affect passivity. Thus they can be used as a minimum solution for connecting a human directly to the controller presented in the previous section.

3.3 Performance Comparison of control strategies

To evaluate the presented controllers in an objective way, they are implemented in *Simulink* and compared in terms of:

- Trajectory tracking
- Dynamic behaviour
- Internal Forces

3.3.1 Internal impedance control with feed-forward of the object dynamics

De Pascali et al. [DPEZ+15] present combination of impedance control on manipulator/object level and feed-forward object dynamics. The internal impedance control relation (between object and manipulator) encompasses a spring and a parallel damper, inertia is used to feed-forward the desired acceleration:

$$M_i\ddot{x}_i^d + D_i(\dot{x}_i^d - \dot{x}_i) + K_i(x_i^d, x_i) = h^x \tag{3.7}$$

I know I have to use consistent notation and explain the variables, comes later ;)
This avoids the necessity of either measuring manipulator acceleration or contact

force. Object dynamics is represented with a feed-forward term, mapped to the manipulators with a weighted pseudoinverse G^+ of the grasp matrix:

$$h^d = G^+(M_o\ddot{x}_o^d + C_o\dot{x}_o^d + g_o) \tag{3.8}$$

Note that this term is not an impedance relation and does not adjust if the environment hinders motion. The combined control law is $h^\Sigma = h^x + h^d$.
The set-up consists of four manipulators, distributed symmetrically around the object. In the first case translation in x-direction commanded. Results in Fig. **??** show good tracking behaviour: no position errors in steady state and only small deviations from the desired values during transient phase. No internal stress is exerted on the object. Due to fast translation high manipulator forces occur.
In the second test case the object is rotated around the z-axis. This is done at a significantly lower speed of at most 1 rad/s, thus the manipulator forces are smaller, desired and actual object trajectory cannot be distinguished in Fig. **??**. However some small internal forces can be seen. Interestingly they are proportional to the simulation step size (running *Simulink's ode3* solver), i.e. a ten-times smaller step size gives ten-times smaller internal forces. Internal forces are calculated based on the geometry in the last simulation step. The correlation between step size and values indicates that these forces are rather due to the discrete nature of the simulation than of the control law generating internal stress. Simulation of the constrained system dynamics as well as calculation of internal wrench is done as described in [**?**].

3.3.2 Internal and external impedance based reference trajectory generation

Caccavale and Villani [CV01] combine both internal and external impedance control. With external we mean a compliant relation between object and (external) environment. The architecture is cascaded, consisting of a two level reference trajectory generation and a motion control loop below. On top-level an impedance relation between object and environment is used to generate a compliant trajectory subject to environmental forces:

$$\alpha M_o(\ddot{x}_o^d - \ddot{x}_o^r) + D_o(\dot{x}_o^d - \dot{x}_o^r) + K_o(x_o^d, x_o^r) = h_{env} \tag{3.9}$$

The constant α scales the object inertia proportionally to a desired value. The control output is the reference object acceleration $\ddot{x}_o^r$, h_{env} is an input. This is sometimes called admittance control, admittance being the inverse of impedance. h_{env} has to be known, but is not easily measured in a practical set-up. Recalling (**??**) the environmental forces can be expressed as:

$$h_{env} = M_o\ddot{x}_o^r + C_o\dot{x}_o^r + g_o - G^\dagger h \tag{3.10}$$

Herein $G^\dagger$ is a generalized inverse of the grasp matrix, selecting the motion inducing components from the measured contact wrench h. $\dot{x}_o^r, x_o^r$ are calculated from $\ddot{x}_o^r$ by integration. From the compliant object trajectory $(\ddot{x}_o^r, \dot{x}_o^r, x_o^r)$ the desired trajectories of the manipulator $(\ddot{x}_i^d, \dot{x}_i^d, x_i^d)$ using the kinematic constraints. The reference manipulator trajectory, enforcing compliant behaviour between manipulators and object, is calculated from manipulator dynamics and internal forces:

$$M_i(\ddot{x}_i^d - \ddot{x}_i^r) + D_i(\dot{x}_i^d - \dot{x}_i^r) + K_i(x_i^d, x_i^r) = VV^\dagger h \tag{3.11}$$

The control output is the reference acceleration of the i-th manipulator $\ddot{x}_i^r$, $\dot{x}_i^r, x_i^r$ are obtained from integration. These variables are the inputs the inner motion control loop (PD-type). The strategy of compliant trajectories allows for high gains in the motion controller. Knowledge of object dynamics and measurement of the contact wrenches is required.
Results for translational motion (see Fig. **??**) are very similar to that of the previous control scheme.
When it comes to rotation, higher internal forces can be observed in Fig. **??**. In this case they are not influenced by numerical parameters of the simulation.
This architecture in contrast to the previous makes use of measured contact wrenches. Contact wrench as measured can be obtained from the constrained system simulation. As described in [CM08] this wrench is than decomposed by kineostatic filters in internal and external components. In this simulation this not cancel out undesired internal stress but magnifies it: when the contact wrench is not fed back and set to zero in the internal force impedance controller, results are slightly better. Note that the simulation represents an ideal case, where all parameters are exactly known and no deviations in grasp positions occur. Behavior in a real experiment may be different and this observation does not mean that the kineostatic-filtered feed-back of contact wrench is unjustified in general.

3.3.3 Intrinsically Passive Controller (IPC)

Introduced by Stramigioli [Str01b] and implemented by Wimböck et al. [WOH08], the architecture has been detailed throughout this work. In contrast to Stramigioli dampers are used in parallel with the manipulator springs and in contrast to Wimböck all springs have 6-DoF. Simulation results for pure translation can be seen in Fig. **??**. The object trajectory falls slightly behind the reference input, the dynamic behaviour is inferior to the previous approaches. Despite very stiff springs the magnitude of force seen in the previous approaches is not reached. No internal forces can be observed.
When it comes to rotation (see Fig. **??**) again the dynamic behaviour falls short of the two other approaches, while the manipulator wrench is higher. Significant internal wrenches are present, they amount up to 50% of the manipulator wrenches.

Chapter 4

Conclusion

List of Figures

1.1 Demonstration of MHI MEISTeR at Fukushima Daiichi NPS 6

2.1 port-Hamiltonian system structure 13
2.2 Variable rest-length spring 19

Bibliography

[BH96] R.G. Bonitz and T.C. Hsia. Internal force-based impedance control for cooperating manipulators. *Robotics and Automation, IEEE Transactions on*, 12(1):78–89, 1996.

[BHM96] M. Buss, H. Hashimoto, and J.B. Moore. Dextrous hand grasping force optimization. *Robotics and Automation, IEEE Transactions on*, 12(3):406–418, 1996.

[CCMV08] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani. Six-dof impedance control of dual-arm cooperative manipulators. *Mechatronics, IEEE/ASME Transactions on*, 13(5):576–586, 2008.

[CM08] F. Caccavle and Uchiyama M. Cooperative Manipulation. In Bruno Siciliano and Oussama Khatib, editors, *Springer Handbook of Robotics*, chapter 29, pages 701–718. Springer Berlin Heidelberg, May 2008. ISBN 978-3-540-23957-4.

[CV01] F. Caccvale and L. Villani. An impedance control strategy for cooperative manipulation. *Advanced Intelligent Mechatronics, IEEE/ASME International Conference on*, 1:343–348, 2001.

[DPEZ+15] L. De Pascali, S. Erhart, L. Zaccarian, F. Biral, and S. Hirche. A decoupling scheme for force control in cooperative multi-robot manipulation tasks. *Manuscript submitted for publication*, 2015.

[DS09] V. Duindam and S. Stramigioli. *Modeling and Control for Efficient Bipedal Walking Robots*. Springer Tracts in Advanced Robotics. Springer-Verlag, Berlin Heidelberg, 2009.

[EH16] S. Erhart and S. Hirche. Model and analysis of the interaction dynamics in cooperative manipulation tasks. *Robotics, IEEE Transactions on*, 2016.

[GFS+14] G. Gioioso, A. Franchi, G. Salvietti, S. Scheggi, and D. Prattichizzo. The flying hand: A formation of UAVs for cooperative aerial telemanipulation. *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 4335–4341, May 2014.

[Goe52] R.C. Goertz. Fundamentals of general-purpose remote manipulators. *Nucleonics*, 10(11):36–45, 1952.

[HKDN13] D. Heck, D. Kostic, A. Denasi, and H. Nijmeijer. Six-dof impedance control of dual-arm cooperative manipulators. *Control Conference (ECC), 2013 European*, pages 2299–2304, July 2013.

[Hog84] N. Hogan. Impedance control: An approach to manipulation. *American Control Conference*, pages 304–313, June 1984.

[Hsu93] P. Hsu. Coordinated control of multiple manipulator systems. *Robotics and Automation, IEEE Transactions on*, 9(4):400–410, 1993.

[HTL00] L. Han, J.C. Trinkle, and Z.X. Li. Grasp analysis as linear matrix inequality problems. *Robotics and Automation, IEEE Transactions on*, 16(6):663–674, 2000.

[LS05] Dongjun Lee and M.W. Spong. Bilateral teleoperation of multiple cooperative robots over delayed communication networks: Theory. *Robotics and Automation (ICRA), International Conference on*, pages 360–365, April 2005.

[LTD14] Mitsubishi Heavy Industries LTD. "meister" remote control robot completes demonstration testing at fukushima daiichi nuclear power station, 2014. Press Information 1775, February 20, 2014; Online, accessed January 13, 2016. URL: `https://www.mhi-global.com/news/story/1402201775.html`.

[MT93] M.J. Massimino and Sheridian T.B. Sensory substitution for force feedback in teleoperation. *Presence, MIT Press Journals*, 2(4):344–352, 1993.

[NPH08] G. Niemeyer, C. Preusche, and G. Hirzinger. Telerobotics. In Bruno Siciliano and Oussama Khatib, editors, *Springer Handbook of Robotics*, chapter 31, pages 741–757. Springer Berlin Heidelberg, May 2008. ISBN 978-3-540-23957-4.

[NS04] G. Niemeyer and J.J.E. Slotine. Telemanipulation with time delays. *International Journal on Robotics Research*, 23(9):873–890, Sep 2004.

[SC92] S.A. Schneider and R.H. Cannon. Object impedance control for cooperative manipulation: theory and experimental results. *Robotics and Automation, IEEE Transactions on*, 8(3):383–394, 1992.

[SD01] S. Stramigioli and V. Duindam. Variable spatial springs for robot control applications. *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, 4:1906–1911, 2001.

[She92] T.B. Sheridian. *Telerobotcis, Automation and Human Supervisory Control.* MIT Press, Cambridge, MA, 1992.

[SMA99] S. Stramigioli, Claudio Melchiorri, and S. Andreotti. A passivity-based control scheme for robotic grasping and manipulation. *Decision and Control, 1999. Proceedings of the 38th IEEE Conference on*, 3:2951–2956, 1999.

[SMH15] D. Sieber, S. Music, and S. Hirche. Multi-robot manipulation controlled by a human with haptic feedback. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2440–2446, Sep 2015.

[SMP14] S. Scheggi, F. Morbidi, and D. Prattichizzo. Human-robot formation control via visual and vibrotactile haptic feedback. *Haptics, IEEE Transactions on*, 7(4):499–511, Oct 2014.

[SSVdSF05] S. Stramigioli, C. Secchi, A.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.

[Str01a] Stefano Stramigioli. Geometric modeling of mechanical systems for interactive control. In Alfonso Banos, Francoise Lamnabhi-Lagarrigue, and FranciscoJ. Montoya, editors, *Advances in the control of nonlinear systems*, volume 264 of *Lecture Notes in Control and Information Sciences*, pages 309–332. Springer London, 2001.

[Str01b] Stefano Stramigioli. *Modeling and IPC Control of Interactive Mechanical Systems: A Coordinate-Free Approach.* Springer-Verlag London, London, UK, 2001.

[Str15] Stefano Stramigioli. Energy-aware robotics. In M. K. Camlibel, A. A. Julius, R. Pasumarthy, and J. Scherpen, editors, *Mathematical Control Theory I*, volume 461 of *Lecture Notes in Control and Information Sciences*, pages 37–50. Springer London, 2015.

[vdS06] Arjan van der Schaft. Port-hamiltonian systems: an introductory survey. In M. Sanz-Sole, J. Soria, J.L. Varona, and J. Verdera, editors, *Proceedings of the International Congress of Mathematicians Vol. III: Invited Lectures*, pages 1339–1365, Madrid, Spain, 2006. European Mathematical Society Publishing House (EMS Ph).

[vdSJ14] Arjan van der Schaft and Dimitri Jeltsema. *Port.Hamiltonian Systems Theory: An Introductory Overview.* Foundations and Trends in Systems and Control. NOW Publishing Inc., Hanover, MA, 2014.

[WKD92] J. Wen and K. Kreutz-Delgado. Motion and force control of multiple robotic manipulators. *Automatica*, 28(4):729–743, 1992.

[WOH06] T. Wimboeck, C. Ott, and G. Hirzinger. Passivity-based object-level impedance control for a multifingered hand. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4621–4627, Oct 2006.

[WOH08] T. Wimboeck, C. Ott, and G. Hirzinger. Analysis and experimental evaluation of the intrinsically passive controller (ipc) for multifingered hands. *Robotics and Automation, IEEE International Conference on*, pages 278–284, May 2008.

[WS08] K. Waldron and J. Schmiedeler. Kinematics. In Bruno Siciliano and Oussama Khatib, editors, *Springer Handbook of Robotics*, chapter 1, pages 9–33. Springer Berlin Heidelberg, May 2008. ISBN 978-3-540-23957-4.

License