Bilateral Telemanipulation With Time Delays: A Two-Layer Approach Combining Passivity and Transparency

Michel Franken, *Student Member, IEEE*, Stefano Stramigioli, *Senior Member, IEEE*, Sarthak Misra, *Member, IEEE*, Cristian Secchi, *Member, IEEE*, and Alessandro Macchelli, *Member, IEEE*

***Abstract*—In this paper, a two-layer approach is presented to guarantee the stable behavior of bilateral telemanipulation systems in the presence of time-varying destabilizing factors such as hard contacts, relaxed user grasps, stiff control settings, and/or communication delays. The approach splits the control architecture into two separate layers. The hierarchical top layer is used to implement a strategy that addresses the desired transparency, and the lower layer ensures that no "virtual" energy is generated. This means that any bilateral controller can be implemented in a passive manner. Separate communication channels connect the layers at the slave and master sides so that information related to exchanged energy is completely separated from information about the desired behavior. Furthermore, the proposed implementation does not depend on any type of assumption about the time delay in the communication channel. By complete separation of the properties of passivity and transparency, each layer can accommodate any number of different implementations that allow for almost independent optimization. Experimental results are presented, which highlight the benefit of the proposed framework.**

***Index Terms*—Bilateral control, passivity, stability, telemanipulation, time delay, transparency.**

I. Introduction

A TELEMANIPULATION chain is composed of a user, a master system, a communication channel, a slave system, and a remote environment upon which the user will act. The master and slave systems both consist of a physical device and a controller (implemented on an embedded system). Typical applications of these chains are the interactions with materials in environments, which are remote, difficult to reach, and/or dangerous for human beings. Bilateral telemanipulation occurs when the user is presented with force information about the interaction between the slave system and the remote environment; see Fig. 1. Such a force feedback is likely to increase the performance of the user with respect to effectiveness, accuracy, and safety in many practical applications, e.g., for robotic surgery, as discussed by Bethea *et al.* [1].

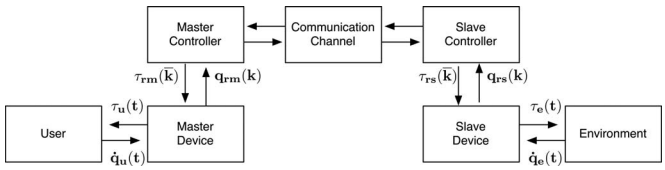

Fig. 1. Schematic overview of a bilateral telemanipulation chain. Both the master and slave devices are impedance-type displays. The information exchanged over the communication channel depends on the implemented controller. τ_* and $\dot{q}_*$ represent torques/forces and velocities, respectively. The subscripts u, rm, rs, and e indicate the interaction between the user and the device, the actuators of the master device, the actuators of the slave device, and the interaction between the slave device and the environment, respectively.

Two important criteria in bilateral telemanipulation are transparency and stability. Transparency is a performance measure of how well the complete system is able to convey to the user the perception of direct interaction with the environment [2]. Many different control algorithms have been proposed in the literature, which try to obtain transparent bilateral teleoperation. Sheridan [3], [4] and Hokayem *et al.* [5] have written extensive survey papers that discuss various approaches to implement bilateral telemanipulation.

Several factors can have a negative influence on the stability of bilateral controllers. Some of these factors are the following:

1) a relaxed grasp of the user;
2) stiff position and force control settings;
3) hard contacts in the remote environment;
4) time delays in the communication channel between the master and the slave.

An elegant solution to prevent these factors from destabilizing the system is found in passivity theory. The interaction between passive systems is guaranteed to be stable, and any proper combination of passive systems will again be a passive system [6]. As the environment can be assumed to be passive and humans can interact very well with passive systems [7], to guarantee passivity of the telemanipulation system itself ensures the stability of the interactions between the user/environment and the telemanipulation system.

An interesting control problem is how to maintain passivity of the telemanipulation chain in the presence of time delays in

Manuscript received November 17, 2010; revised March 18, 2011; accepted April 7, 2011. Date of publication June 7, 2011; date of current version August 10, 2011. This paper was recommended for publication by Associate Editor N. Y. Chong and Editor B. J. Nelson upon evaluation of the reviewers' comments.

M. Franken, S. Stramigioli, and S. Misra are with the Institute for Biomedical Technology and Technical Medicine, Control Engineering Group, University of Twente, 7500 AE Enschede, the Netherlands (e-mail: m.c.j.franken@utwente.nl; S.Stramigioli@utwente.nl; s.misra@utwente.nl).

C. Secchi is with the Department of Sciences and Methods of Engineering, University of Modena and Reggio Emilia, 42100 Reggio Emilia, Italy (e-mail: cristian.secchi@unimore.it).

A. Macchelli is with the Department of Electronics, Informatics and Systems, University of Bolgona, 40136 Bologna, Italy (e-mail: alessandro.macchelli@unibo.it).

Digital Object Identifier 10.1109/TRO.2011.2142430

the communication channel. As the master and slave systems can be located at different sites, it is likely to assume that a certain amount of time delay will be present in the communication channel. Time delays can also occur due to various other processes other than physical distance, e.g., congestion of the network, and the coding and decoding of the signals exchanged through this network between the master and slave systems. Passive control schemes that work in the presence of time delays have been developed, e.g., the scattering and wave variable approaches described by Anderson *et al.* [8] and Niemeyer *et al.* [9]. Arcara *et al.* [10] and Lawn *et al.* [11] have compared several passivity-based algorithms to nonpassive algorithms with respect to stability and the level of transparency that could be achieved for a range of communication delays. Passivity-based approaches are indeed found to be stable in the presence of even significant time delays, but the level of transparency that could be obtained was criticized.

The problem with current passivity-based methods is that they are specifically designed around a certain type of information exchange. This places strict limitations on the rest of the controller. As we will discuss, there are a multitude of control architectures designed for transparency that do not fit within those passivity-based methods. Given the benefits of passivity with respect to guaranteed stability, we want to design a framework in which any controller can be implemented in a passive manner given arbitrary time delays.

In this paper, we will present a new control framework for passive bilateral telemanipulation. The framework is composed of two layers placed in a hierarchical structure. Each layer is further designed for a specific purpose, either to obtain transparency or to maintain passivity. In the top layer, the *transparency layer*, a control structure can be implemented to provide the best possible transparency of the telemanipulation chain, taking into account all available information about the system, the environment, and the task the user is executing. The commands that are computed in this layer are passed to the bottom layer, the *passivity layer*. This layer contains an algorithm to maintain passivity of the total system. The key element of this algorithm is to define two communicating energy storage tanks from which the motions of both the slave and the master are powered. The use of two control layers to combine passivity and transparency and the working of the passivity layer, in which energy is treated in the most general sense possible and completely free of any assumptions on the time delay in the communication channel, are the main contributions of this paper.

In the rest of this paper, an impedance causality for both the master and slave systems (velocities as input and forces as output to the robotic devices) is assumed. For these devices, the energy exchanged with the outside world can be precisely determined, which is a base assumption of the work presented in Section IV. The paper is organized as follows. In Section II, we introduce the concepts of passivity, and the related work will be discussed. In Section III, we further elaborates on the two-layer framework. Section IV contains the theory of the passivity layer. In Section V, we present a full implementation and the experimental results, which were obtained with the proposed framework, and demonstrate their effectiveness. A discussion about the proposed framework in relation to other proposed methods is presented in Section VI. We conclude the paper and provide directions for future work in Section VII.

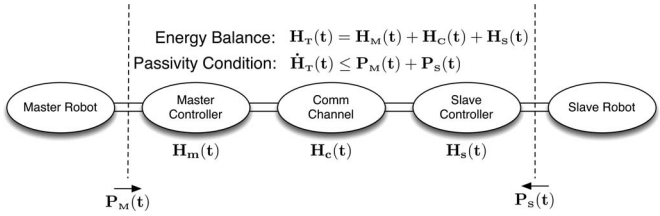

Fig. 2. Energy balance of the telemanipulation chain. The double lines indicate an energetic connection.

II. Passivity and Related Work

As mentioned in Section I, a passive implementation of a bilateral controller ensures stable behavior of the system even in the presence of factors that could otherwise destabilize the system. We now provide a review of the important concepts, which pertains to passive telemanipulation systems that are essential for the derivations presented later in Section IV. In addition, four related approaches will be discussed. Each of these approaches constitutes a contribution to the research field, but in order to facilitate the comparison of those approaches with the framework proposed in this paper, we will indicate factors that can be considered, in the opinion of the authors, as structural limitations.

A system is said to be passive if the energy that can be extracted from it is bounded by the injected and initial stored energy. Any proper combination of passive systems will again be passive [6]. Independent of anything else, including the goal of the system, an energy balance of the telemanipulation system can be composed of the energy present in all of its components. The total energy $H_T(t)$ present in the control system at instant t is

$$H_T(t) = H_M(t) + H_C(t) + H_S(t) \tag{1}$$

where $H_M(t)$, $H_S(t)$, and $H_C(t)$ represents the energy present at the master side, at the slave side, and in the communication channel, respectively. This is shown in Fig. 2. Assuming the initially stored energy zero, the passivity condition of the system is

$$H_T(t) \geq 0. \tag{2}$$

Physical energy exchange during operation takes place between the user and the master system, as well as between the slave system and the environment. The only requirement, therefore, that is necessary to ensure a passive interconnection of the entire system with the physical world is

$$\dot{H}_T(t) \leq P_M(t) + P_S(t) \tag{3}$$

where $P_M(t)$ and $P_S(t)$ are, respectively, the power flowing from the master and slave robot into the master and slave controller, and $\dot{H}_T(t)$ is the rate of change of the energy balance of the system. Equations (2) and (3) ensures passivity of the

system and a passive connection of the system with the physical world, respectively.

A. Scattering/Wave-Variable-Based Approaches

It is well known that the direct exchange of power variables (velocities and forces) between the master and slave devices generates "virtual" energy in the presence of time delays in the communication channel. The scattering and wave variables approaches developed by Anderson *et al.* [8] and Niemeyer *et al.* [9] apply a coding scheme to the power variables to turn the time-delayed communication channel into a passive element. When the controllers at both the master and slave sides are, furthermore, passive, the complete system is passive according to (3); such a complete approach is described by Secchi *et al.* [12].

A wave variable contains both information related to the energy exchange that occurs at that side and the desired behavior to be displayed by the other device. Niemeyer [13] describes a wave variable as a general "move/push" command to be interpreted by the receiving device, and the returning wave describes the response of that device to the received command. This means that the motion performed by the user and the resulting force feedback are separated in time by the round-trip time of the communication channel. Other transparency-related problems arise due to the nature of the (de)coding process and/or nonidealities in the communication channel (time-varying delay and package loss), e.g., position and force mismatch. Extensions to improve the performance include the use of Smith predictors [14], the transmission of wave integrals [9], and the combination of wave variables with the transmission of interaction measurements as discussed by Tanner *et al.* [15].

B. Time Domain Passivity Control

A different solution to the passivity problem was proposed by Ryu *et al.* [16]. There the time domain passivity control (TDPC) algorithm, which was developed by Hannaford *et al.* [17] for passive interaction with virtual environments, was applied to bilateral telemanipulation. The TDPC approach introduces a passivity observer (PO) and a passivity controller (PC). This algorithm enforces (3) with $H_C = 0$ as no communication channel is considered. For this algorithm, simultaneous information about the energy exchange at the master and the slave sides is required and is, as such, not applicable to systems with time delays in the communication channel. Two extensions have been proposed to extend the TDPC approach to the time-delayed situation.

Artigas *et al.* [18] incorporate an energy reference algorithm. Artigas *et al.* [19] further extended this approach to also include a passive coupling between the continuous and the discrete domain. The reference algorithm applies a forward and a backward PO, which estimates the energy in the communication channel based on the locally transmitted and received power variables and an estimate of the fixed transmission delay. At each side of the communication channel, a PC maintains passivity according to the PO at that side.

Ryu *et al.* [20] split the energy interaction into an incoming and outgoing energy flow E_{in} and E_{out}. Each side transmits its E_{in} to the other side where passivity of E_{out} with respect to the received value of E_{in} is maintained by a PC. As the transmitted packets symbolize an amount of energy, the passivity of this approach is perfectly robust against time-varying delays and even packet loss in the communication channel.

These approaches have merged into a single algorithm as proposed by Ryu *et al.* [21]. However, these approaches are not suitable for the implementation of impedance reflection (IR) algorithms, e.g., [22], where the feedback force to the user is predicted based on a local, possibly adaptive, model of the remote environment. The work of Artigas is centered around the transmission of power variables and cannot accommodate the transmission of model parameters. In the algorithm of Ryu *et al.,* the problem is that with an IR algorithm the energy extracted by the user E_{out} at the master side is likely to occur before E_{in} actually occurs at the slave side. This means that the PC at the master side will prevent the computed feedback force to be applied to the user as it would force the PO to become negative. A first approach to use a TDPC algorithm with an IR algorithm was proposed by Kawashima *et al.* [23]. A TDPC structure is used to adapt the locally computed feedback force based on the actual measured but delayed interaction force to make the system passive. This approach requires exact knowledge about the time delay that is present in the communication channel.

C. Energy Bounding Algorithm

Another approach that originates from research toward passive interaction with virtual environments is the energy bounding algorithm (EBA) proposed by Kim *et al.* [24]. Seo *et al.* [25] have applied the EBA to time-delayed bilateral telemanipulation. The EBA limits the generated "virtual" energy to the dissipated energy by friction at the master and slave sides. For this, it uses models of viscous friction in the devices, which are possibly extended with assumptions about the viscous friction in the user's arm and/or environment.

Deviations from the physical friction with respect to the modeled friction can jeopardize stability of the system for which reason a conservative lower bound of the friction needs to be selected. Due to the nature of the derived update rule, as indicated by the authors in [24], the force applied by the control system cannot be adjusted when the devices are perfectly stationary. Finally, it appears, based on [25], that in the bilateral telemanipulation application it can only work when the force exerted by the slave device is used as the feedback force to the user instead of the measured interaction force between the slave device and the remote environment. This can severely limit the achievable transparency in the presence of time delays and limits the implementable bilateral controller to that specific implementation.

D. Passive Set-Position Modulation

A recent approach to deal with bilateral telemanipulation is the passive set-position modulation (PSPM) framework that is proposed by Lee *et al.* [26], [27]. This approach is centered around a spring-damper controller. The energy dissipated by the "virtual" damper is stored in an energy tank. The jump in

spring potential due to the discrete jump of the set position by the control algorithm is limited to the available energy in the tank (a negative jump adds energy to the tank). In the bilateral telemanipulation application, excess energy in the tank is transmitted to the other side or is dissipated.

There are several issues related to the working of the PSPM. Most notably, the underlying assumption is that part of the control system can be regarded as continuous time. The set-position signal is a discrete signal, but the position of the device, as used in the servo control loop, is considered as a continuous signal. However, the control system is always sampled even though the update rate for the set position might be much lower than the fundamental sampling rate of the servo control loop. It is well known that the description of a passive element in continuous time can generate energy when implemented on a discrete medium. Therefore, an extended form of Colgate's passivity condition [28] that relates the parameters of the controller, the sample frequency, and the device friction is necessary to guarantee passivity of the system. By the assumption of a constant sampling time ΔT_S, the condition described in [26] becomes

$$B_{\mathrm{dev}} \geq 2B_C + \frac{K_C \Delta T_S}{2} \tag{4}$$

where B_{dev}, B_C, and K_C indicate the physical viscous device friction and the implemented viscous damping and stiffness in the PSPM element, respectively. Equation (4) states that the required physical viscous damping has to be at least twice as large as the implemented virtual damping for the system to be guaranteed to be passive.

The input to the controller is a set position for the spring. This means that bilateral control algorithms that compute a desired force to be applied to the device(s) require intermediate data processing. This data processing transforms a desired control force into the required set position. This appears elaborate and noise sensitive due to the inherent presence of the velocity estimate, and it requires the set-position signal to be updated at the same frequency as the velocity estimate. This last factor degrades the validity of the assumption that the servo control loop can be considered to be in continuous time.

Finally, the PSPM relies on the use of a constant viscous damper to extract energy into the energy tanks. This means that the response will already always be damped, even when there is enough energy in the tank. The excess extracted energy is artificially dissipated by thresholding the level of the energy tank. This constant damping also needs to be taken into account in any higher level control architecture that is connected through intermediate data processing to prevent an over-damped response of the system. With the PSPM, it is, therefore, difficult to separate the design of the controller to display the desired behavior and the manner in which passivity is maintained.

III. Proposed Two-Layer Framework

In the previous section, several passivity-based control structures were discussed. Without making any assumptions about the type of controllers implemented, we can formulate the control goals of a passive bilateral telemanipulation system as follows. The slave device needs to display the behavior desired by the user, and the master device needs to accurately provide force feedback about the interaction between the slave device and the remote environment, unless this behavior violates the passivity condition of the telemanipulation system.

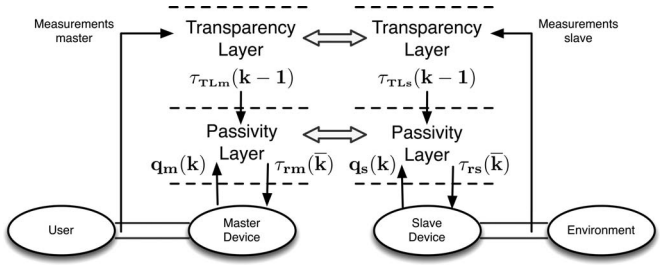

Fig. 3. Two-layer algorithm for bilateral telemanipulation: Double connections indicate an energetic interaction.

This shows that a natural layering in control objectives arises. First, a desired control action needs to be computed so that the master and slave devices display the desired behavior/information. Then, a "check" is to be performed of how this desired action will influence the energy balance of the system. If passivity will not be violated, it can directly be applied to the physical system, but if passivity is expected to be lost due to the desired control action, it should be modified before application to the physical system. Such an approach allows for the highest possible transparency given that passivity needs to be preserved.

This natural layering can also be directly transformed into a control structure. An algorithm that combines transparency and passivity in the discussed manner would be a two-layer structure, as shown in Fig. 3. The transparency layer contains a control algorithm to display the desired behavior and obtain transparency. Ideally, this could be any type of bilateral control algorithm. The only requirement that the framework which is presented in this paper places on the implemented controller is that it computes forces to be applied to the master and slave devices. In order to compute the desired control action $\tau_{\mathrm{TL}*}(k)$, access is required to a specific part of the measured interaction data, e.g., forces, positions, and/or velocities, where m and s instead of $*$ indicate the master and the slave, respectively. The passivity layer on the other hand monitors and enforces the energy balance of the system according to the algorithm discussed in Section IV.

The benefit of the strict separation into layers is that the optimization of the strategy used to ensure optimal transparency does not depend on the strategy used to ensure passivity and vice versa. As passivity does not have to be considered in the design of the transparency layer, the whole range of control techniques which are nonpassive, e.g., most filtering techniques, can be applied without problems. In addition, due to this strict separation into layers two two-way communication channels between the master and slave systems can be defined. One channel is used to communicate energy-exchange-related information between the passivity layers and the second channel to communicate information related to the desired behavior to be displayed by the devices between the transparency layers. This means that no (de)coding process is required as with wave-variable-based

approaches. Furthermore, no restrictions are necessary on the information exchanged between the transparency layers.

It should be noted that of the passivity-based control structures that are listed in the previous section, the time-delayed TDPC approach by Ryu *et al.* [20], [21] and the PSPM framework by Lee *et al.* [26] can also be represented as a two-layer framework, as depicted in Fig. 3. However, as mentioned in the previous section, the implementation of the time-delayed TDPC approach restricts the types of bilateral controllers that can be implemented (no IR algorithms) and, furthermore, only acts upon a loss of passivity and was not intended to shape the interaction to prevent a loss of passivity. The time-delayed TDPC approach is centered on the energy exchange that occurs in the communication channel. Any augmentation of the force feedback to the user in order to improve his performance during the execution of a task, e.g., by the incorporation of virtual fixtures as described by Abbott *et al.* [29], needs to be separately implemented with additional measures to ensure stability [28]. The PSPM approach can accommodate these features by means of the intermediate data processing capabilities but requires an additional hardware–controller settings condition to be satisfied. Furthermore, although the intermediate data processing capability is there, its implementation might not be straightforward. In the next section, we will introduce an implementation of the passivity layer that is in the opinion of the authors free of such limitations.

IV. Passivity Layer

In this section, we will discuss how the passivity layer that was introduced in the previous section works.[1] The only thing that is needed to know about the transparency layer is that it generates desired torques to be applied to the master and slave devices.

Assume that the slave device is operating under position control of the master device. Every movement that the slave device makes will have an associated energetic cost. In order for the system to be passive, this amount of energy will have to be present at the slave side at the moment the movement is executed. The passivity condition of (2), which is applied to the energy balance of the system (1), also requires that that same amount of energy will have had to be injected previously by the user at the master side and to be transported to the slave side through the communication channel. Depending on the implemented bilateral control algorithm, the same can apply in reverse to energy extraction at the master side. This clearly requires the transport of energy between the master and the slave systems.

Due to the time delays, which separates the master and slave systems, it is not possible to simultaneously monitor the energy exchange at both interaction ports. This means that when the user commands a motion to be executed by the slave, it is not known *a priori* (exactly) how much energy is required by the slave device to execute that motion. To this end, the concept of a lossless energy tank is introduced in the passivity layer at both the master and the slave sides, which can exchange energy. The level of these tanks can be interpreted as a tight energy budget from which controlled movements can be powered and which are being replenished by the user at the master side when necessary or if possible/desired also at the slave side. If the energy level in the tanks is low, the controlled movements that the system can make are restricted. An extreme situation occurs when the tank is completely empty in which situation the system cannot make a controlled movement at all. Passivity will always be maintained as all the energy present in the system has been injected by the user, and each system cannot use more energy than is available in its energy tank.

Adjustments made by the passivity layer to the commands of the bilateral controller, which are implemented in the transparency layer, can have a negative influence on the achievable transparency by the telemanipulation system. This decrease, however, is minimized to the point, where passivity is maintained and, thus, stable behavior guaranteed.

In the following subsections, the four components of the passivity layer at each side are discussed. As these operations are implemented at both sides in the same manner, subscripts that indicate the master and slave have been omitted for now. In order to illustrate the working of the passivity layer, a flow chart of all the steps in the passivity layer for either side of the telemanipulation system is presented in Fig. 6 at the end of this section.

A. Monitoring Energy Flows

At both the master and the slave sides, the following three energy flows can be identified:

1) an energy exchange with the physical world;
2) an energy flow to the other system;
3) an energy flow from the other system.

We will now show how each of these flows can be monitored and regulated in order to maintain passivity according to (2) and (3).

On the master and slave sides, the controllers will have to control two robots, which will interact with the user and the environment. As the controller is implemented on some sort of embedded processing unit, there is a connection between the continuous and discrete domains. Let $\dot{q}(t)$ represent the velocity vector of the actuators at time t and $q(k)$ the sampled position vector of the actuators at sample instant k. Consider the sample period to be $\overline{k}$. The torques exerted by the actuators on the robot during sample period $\overline{k}$ is given by $\tau_r(\overline{k})$, which is held constant during the sample interval. Thus, the energy exchange between the discrete time controller and the physical world $\Delta H_I(k)$, during the sample interval between the time instants $k-1$ and k, is

$$\begin{aligned}\Delta H_I(k) &= \int_{(k-1)\Delta T_S}^{k\Delta T_S} \tau_r(\overline{k})\dot{q}(t)dt \\ &= \tau_r(\overline{k})(q(k)-q(k-1)) \\ &= \tau_r(\overline{k})\Delta q(k) \end{aligned} \tag{5}$$

[1]The index k is used to indicate instantaneous values at the sampling instant k, and the index $\overline{k}$ is used to indicate variables related to an interval between sampling instants $k-1$ and k.

where ΔT_S is the length of the sample period, and $\Delta q(k)$ is the computed position difference at sample instant k that occurred during the sample period $\overline{k}$. Therefore, only a position measurement is required to determine the energy exchange, which was introduced by Stramigioli *et al.* [30]. The computation of (5) assumes a perfect servo loop and noise-free position measurement. If bounds can be derived for the inaccuracies in both the servo loop and the measurement, (5) can be adjusted to account for these imperfections. As (5) only holds for impedance-type systems (force out causality), we require the entire control structure and, thus, the transparency layer to adhere to this causality.

As far as the energy exchange between the master and slave is concerned, we can consider the possibility to send energy quanta from the master to the slave when energy is available in the energy tank at the master side and vice versa. These quanta can be transmitted in the form of packets that contain the amount of energy send. Several possible communication protocols for this energy transfer will be discussed in Section IV-C. Both master and slave can implement completely asynchronously the following operations (6)–(8). When such an energy packet arrives at the other side, it is stored in a receiving queue:

$$H_+(k) = \sum_{i \in Q(k)} \bar{H}(i) \tag{6}$$

where $Q(k)$ represents the set of all energy packets that are present in the receiving queue of the master at sample instant k, and $\bar{H}(i)$ represents the ith energy packet. Therefore, $H_+(k)$ represents the total amount of energy that is present in the receiving queue at that time instant. At each sample instant k, the receiving queue is emptied, which means that the energy present in the receiving queue $H_+(k)$ is added to the level of the energy tank. The exchanged energy with the physical world during the previous sample period is computed according to (5) and subtracted from the level of the energy tank. The energy level of the tank after these operations $H(k)$ is

$$H(k) = H(\overline{k}) + H_+(k) - \Delta H_I(k) \tag{7}$$

where $H(\overline{k})$ is the energy level of the tank before the operations at sampling instant k. Based on the chosen energy transport protocol, an energy quantum $H_-(k)$ is determined to transmit to the other side. This energy quantum is at least limited to $H(k)$ to preserve passivity. The amount of energy that is transmitted is extracted from the energy tank. The energy that is left in the tank after these operations and, thus, available during the next sampling period $H(\overline{k+1})$ is

$$H(\overline{k+1}) = H(k) - H_-(k). \tag{8}$$

With this algorithm, we are, therefore, able to compute the exact energy balance at each instant of time when sampling occurs, and passivity according to (1)–(3) is guaranteed. The level of the energy tanks is the total energy present on the master and slave sides H_M and H_S, respectively. The sum of all the energy packets in the communication channel gives the total energy present in the communication channel H_C. A graphical representation of (5)–(8) is given in Fig. 4, which indicates the two steps of the energy flow computation.

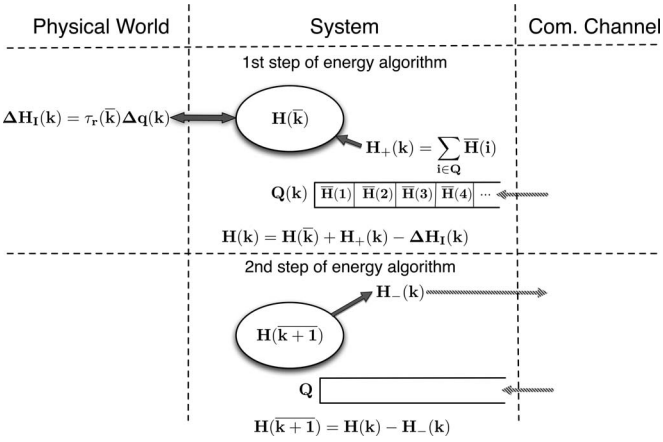

Fig. 4. Processing energy flows. Energy received out of the communication channel is added to the level of the energy tank and the energy exchanged with the physical world is subtracted from the energy level (first step). An energy packet is transmitted to the other system (second step). The double arrow indicates that the energy exchange with the physical world can both be positive and negative.

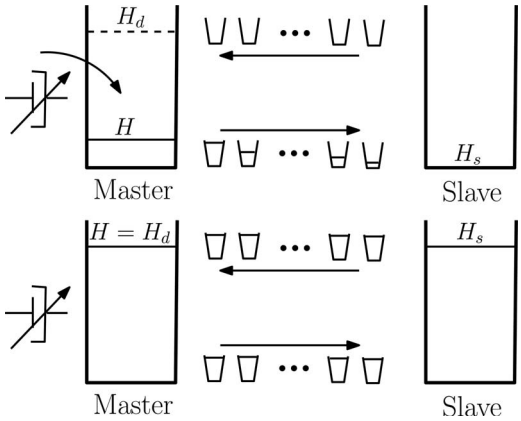

Fig. 5. Level synchronization between energy tanks. Modulated damper extracts energy from the master, and the implemented energy transport protocol forces the energy level in the master and slave tanks to synchronize.

As each packet represents an amount of energy, the passivity of the communication channel is unaffected by any nondeterministic time delay, similar to [21] and [26]. The change of energy $\Delta H_C(k)$ in the communication channel at the sample instant k can be expressed as

$$\Delta H_C(k) = H_{-M}(k) - H_{+M}(k) + H_{-S}(k) - H_{+S}(k) \tag{9}$$

where $H_{-M}(k)$ and $H_{+M}(k)$ are the energy flow into and from the communication channel at the master side, and $H_{-S}(k)$ and $H_{+S}(k)$ are the energy flows at the slave side. The total energy in the communication channel $H_C(k)$ is

$$\begin{aligned} H_C(k) &= \sum_{i=1}^{k} \Delta H_C(k) \\ &= \sum_{i=0}^{k} H_{-M}(i) - H_{+M}(i) + H_{-S}(i) - H_{+S}(i). \end{aligned} \tag{10}$$

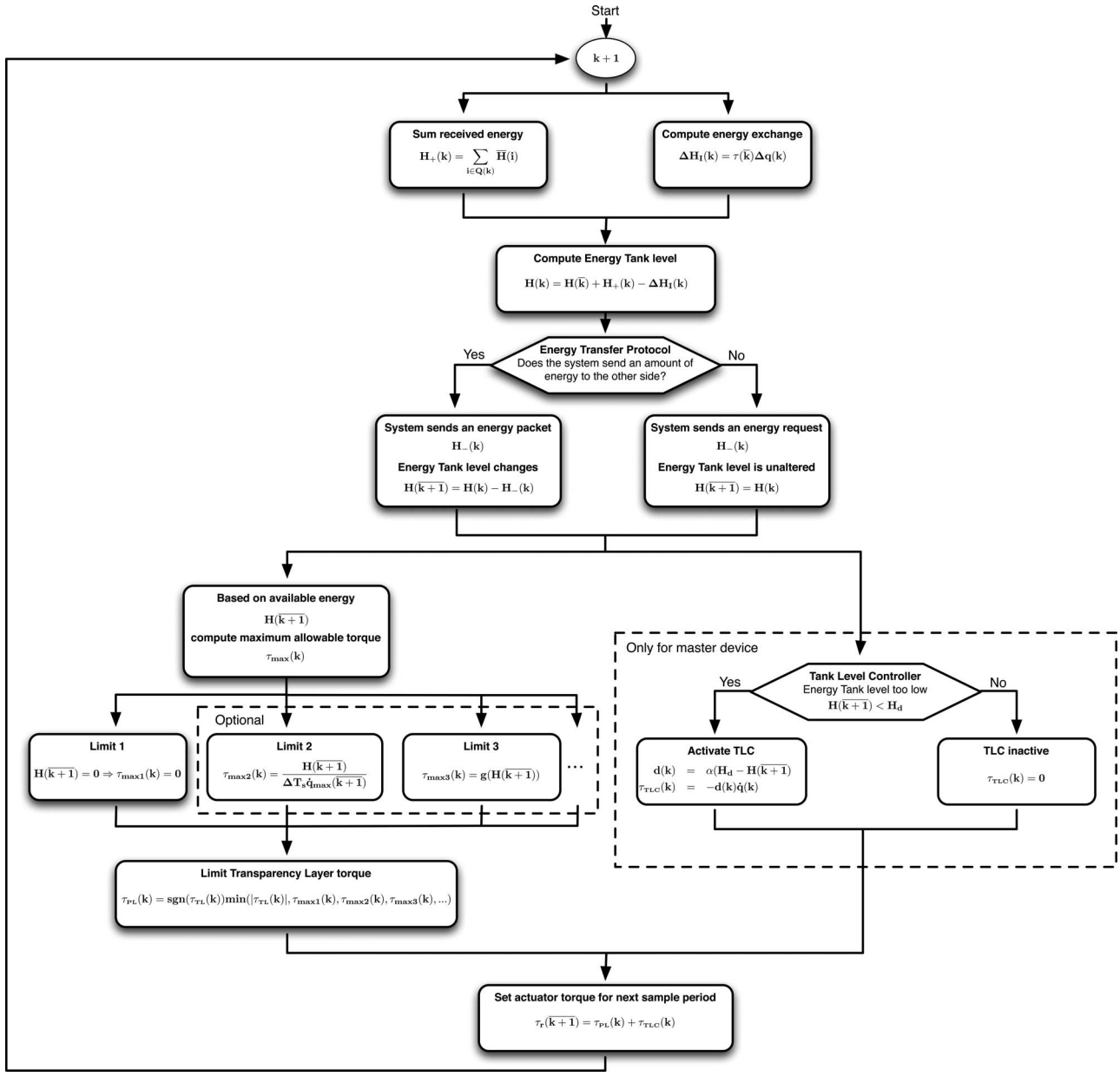

Fig. 6. Workflow of the complete passivity layer at either side of the telemanipulation system. Workflow assumes that $\tau_{\mathrm{TL}}(k)$ has already been computed. First, the incoming energy flows are evaluated. Afterward, the energy flow toward the other system is computed and handled. Finally, the limiting values for the torque that originates from the transparency layer are computed. For the master system, the TLC is activated, if necessary. The limited transparency layer torque and TLC torque combined form the feedback force to the user for the next sampling period.

Due to the time delay in the communication channel

$$\begin{aligned} H_{-S}(i) &= H_{+M}(i + d_{SM}(i)) \\ H_{-M}(i) &= H_{+S}(i + d_{MS}(i)) \end{aligned} \tag{11}$$

where $d_{SM}(i) \geq 0$ and $d_{MS}(i) \geq 0$ represent the possibly non-deterministic time delays in the communication channel, which include possible package loss. Therefore

$$\begin{aligned} \sum_{i=0}^{k} H_{-M}(i) &\geq \sum_{i=0}^{k} H_{+S}(k) \\ \sum_{i=0}^{k} H_{-S}(i) &\geq \sum_{i=0}^{k} H_{+M}(k) \end{aligned} \tag{12}$$

so that

$$H_C(k) \geq 0 \quad \forall k \tag{13}$$

which means that the communication channel can never produce energy as long as packet duplication is prevented. Duplicated packets can easily be handled by the inclusion of a timestamp in each packet.

B. Energy Tanks

In the previous section, we have shown that there exist three energy flows at both the master and the slave sides. The desired control actions that are determined by the transparency layer will influence the energy exchange with the physical world and, thus,

how much energy is flowing into or out of the total system. In order to completely separate the passivity layer from the transparency layer, a method is required to regulate the energy level independent of what the transparency layer is commanding.

To this end, a tank level controller (TLC) is defined in the passivity layer at the master side. The function of this TLC is to monitor the energy level of the local tank $H_M(\overline{k+1})$, with respect to a desired level H_D. Whenever $H_M(\overline{k+1})$ is lower than H_D at a sampling instant k, the TLC extracts a small additional amount of energy from the user during the next sampling period $\overline{k+1}$ to replenish the tank. The usage of such a TLC will enable the control architecture to always recover from a deadlock situation in a passive manner when all the energy stored in the system is depleted.

Several TLC implementations are possible. In this paper, the TLC is a modulated viscous damper, which applies a small opposing torque $\tau_{\mathrm{TLC}}(k)$ to the user's movement to extract energy from the user into the energy tank:

$$\tau_{\mathrm{TLC}}(k) = -d(k)\dot{q}_m(k)$$
$$d(k) = \begin{cases} \alpha(H_D - H_M(\overline{k+1})), & \text{if} \quad H_M(\overline{k+1}) < H_D \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

where α is a parameter that can be used to tune the rate at which energy is extracted from the user, and $\alpha > 0$. If α is set to a high value and/or the user moves very fast, an overshoot of the energy level in the system with respect to the desired energy level can occur. The values to be set for α and H_D are highly dependent on the device characteristics.

It is important to note that although this strategy might appear similar at first glance to the TDPC strategy by Ryu *et al.* [16], its purpose is in fact very different. The PC element in the TDPC algorithm is used to dissipate virtually generated energy, whereas in this application the modulated damper is primarily activated to make energy available in the system. It should also be noted that the presented strategy is only one way to extract energy from the user and that the framework can accommodate many alternatives.

C. Energy Transport

The TLC will make energy available at the master side, but energy is required at the slave device for it to be able to perform its task. In this section, various protocols are discussed that can be implemented to regulate the distribution of energy through the system, which range from simple open loop protocols to more complex protocols.

1) Simple Energy Transfer Protocol: Energy can be distributed through the system by the use of the simple energy transfer protocol (SETP). Both the master and slave systems transmit a fixed fraction β of its energy level (when energy is available) to the other system. This will cause the total energy in the system to be distributed over the master and slave systems and the communication channel. As this is a bilateral transfer protocol, it is not dependent on where the energy is entering the system. This is illustrated in Fig. 5, where the energy tanks are depicted as water barrels, each packet as a glass, and the energy quanta of each packet as the water level inside the glass.

When the system can be described as a discrete linear time invariant (LTI) system, it can be proven, mathematically, that the energy levels in the two tanks will converge to the same value when there is no interaction with the physical world, irrespective of the initial energy distribution. The LTI model Σ_1 of the system is

$$\Sigma_1 : x(\overline{k+1}) = \begin{bmatrix} 1-\beta & 0 \\ 0 & 1-\beta \end{bmatrix} x(\overline{k}) + \begin{bmatrix} 0 & \beta \\ \beta & 0 \end{bmatrix} x(\overline{k-d}) \tag{15}$$

where $x = \begin{bmatrix} H_M \\ H_S \end{bmatrix}$, and d indicates the constant communication delay, respectively. From Σ_1, a new state H_{dif} is derived that describes the dynamics of the difference between the tank levels:

$$\begin{aligned} H_{\mathrm{dif}}(\overline{k+1}) &= H_M(\overline{k+1}) - H_S(\overline{k+1}) \\ &= (1-\beta)H_{\mathrm{dif}}(\overline{k}) - \beta H_{\mathrm{dif}}(\overline{k-d}). \end{aligned} \tag{16}$$

The characteristic polynomial $P(z)$ that describes the dynamic behavior of just this new state is described as [31]

$$P(z) = z^{d+1} - (1-\beta)z^d + \beta. \tag{17}$$

The investigation of the stability of this system, without explicit computation of the roots of the polynomial, can be performed by the use of the Jury stability criterion [32]. This criterion states that if certain terms that are computed from the coefficients of the polynomial are positive, the system is asymptotically stable. Application of this criterion to $P(z)$ indicates that the following terms have to be positive:

$$\begin{aligned} &\frac{4\beta(1-\beta)}{d\beta+1} > 0 \\ &\frac{-r\beta^2 + (r-1)\beta + 1}{(r-1)\beta+1} > 0 \quad \forall r \in [0..d-1]. \end{aligned} \tag{18}$$

For $0 < \beta < 1$ and any d, all the terms of (18) are positive. This indicates that the tank level difference by the use of the SETP is asymptotically stable and will converge to zero in the absence of external inputs, albeit that the settling time can be extremely large for large d and/or β. If the time in which the TLC extracts the energy to fill the master tank is small compared with the settling time of the SETP, an overshoot of the energy level of the tanks with respect to H_D can occur.

This derivation of asymptotic stability also holds for systems Σ_2, with asynchronous delays as long as the total number of delay states n is even. Such a system has the same characteristic polynomial $P_1(z)$ as Σ_1, given that $n = 2d$:

$$P_1(z) = z^{n+2} - 2(1-\beta)z^{n+1} + (1-\beta)^2 z^n - \beta^2. \tag{19}$$

The asymptotic stability of Σ_1 implies the asymptotic stability of Σ_2.

2) Advanced Energy Transfer Protocols: The SETP implies that energy quanta are continuously being exchanged between the master and slave systems. This indicates that the user besides filling both tanks with energy will also have to saturate the communication channel with energy packets. For Σ_1,

there are $2d$ energy packets in the communication channel. Assume that the TLC has extracted precisely enough energy to let the energy level in both tanks converge to H_D, and this convergence has taken place. In that situation, each energy packet in the communication channel has the same value βH_d. Therefore, in this situation, the total amount of energy in the communication channel $\overline{H}_C$ is

$$\overline{H}_C = 2d\beta H_D. \tag{20}$$

The total amount of energy in the communication channel can, therefore, become quite large for larger time delays and/or β.

More complex transfer protocols can be implemented, each with its own specific benefits and drawbacks. A transfer protocol that is still simple, but does not have constant energy exchange between master and systems, is to change the positive energy quanta being sent from the slave to the master into energy requests. The transfer protocol at the master side will then send an initial amount of energy to the slave side to fill the tank and the slave will only send energy requests to the master when the level in the tank drops below that desired level due to energetic interaction with the physical world. The master side records the total energy request by the slave and will send a percentage of its available energy toward the slave until the energy request is satisfied. A drawback of this protocol is that the energy request and the subsequent delivery are separated in time by the round-trip time of the communication channel. This will have to be taken into account when selecting the desired energy level of both tanks.

Now assume that an IR algorithm is implemented in the transparency layer. As the interaction forces are now predicted at the master side, it is possible to record the energy exchange and transmit this energy directly to the slave side. The energy tanks are then solely used to deal with model inaccuracies and the time delays in the communication channel.

This shows that although the transparency layer and passivity layer are completely separated and can be tuned independently, the energy transfer protocols that can be implemented in the passivity layer are restricted by the chosen implementation of the transparency layer.

D. Saturation of Controlled Torque

The transparency layer computes a controlled torque $\tau_{\mathrm{TL}}(k)$ at each side, which is to be applied to the master and slave devices during the sampling period $\overline{k+1}$ to display the desired behavior. At both sides, the passivity layer enforces limits on this desired torque in order to maintain passivity. The resulting limited torque $\tau_{\mathrm{PL}}(k)$ will be applied to the actuators during the sample period $\overline{k+1}$. It should be noted that although $H(\overline{k+1})$ indicates the energy level in the tank during the sample period $\overline{k+1}$, its value is known after the procedure of Section IV-A has been performed at the sample instant k. This procedure is performed before $\tau_{\mathrm{PL}}(k)$ is computed so that the value of $H(\overline{k+1})$ can be used to compute $\tau_{\mathrm{PL}}(k)$.

The fundamental limit that the passivity layer enforces is that when no energy is available at a side, the controlled torque that can be applied at that side during the coming sampling period is zero:

$$\tau_{\max 1}(k) = \begin{cases} 0, & \text{if} \quad H(\overline{k+1}) \leq 0 \\ \tau_{\mathrm{TL}}(k), & \text{otherwise.} \end{cases} \tag{21}$$

Between two sample instants, there is no way to detect, act upon, and, therefore, prevent a possible loss of passivity. It is, however, possible to minimize the chance of such a loss by the implementation of additional saturation functions. We know that the interval before a next sample will last ΔT_S s and suppose that the device is moving with velocity $\dot{q}(\overline{k})$. If the force applied during sample period $\overline{k+1}$ were to have a relatively small influence on the velocity with which the device is moving, an initial estimate of the energy exchange that will occur would be

$$\Delta H_I(\overline{k+1}) = \tau(\overline{k+1})\dot{q}(\overline{k})\Delta T_S \tag{22}$$

and therefore, an upper bound for $\tau(\overline{k+1})$ to limit this energy exchange to the available energy would be

$$\tau_{\max 2}(k) = \frac{H(\overline{k+1})}{\dot{q}(\overline{k})\Delta T_S}. \tag{23}$$

The applied force during $\overline{k+1}$, however, will most often influence the velocity with which the device is moving. This influence might be approximated based on a competent dynamic model of the system so that the worst-case velocity of the system can be expressed as a function of the applied force, the current velocity, and the duration of the sample period $\dot{q}_{\max}(\tau_{\mathrm{PL}}(\overline{k+1}), \dot{q}(\overline{k}), \Delta T_S)$. $\dot{q}_{\max}$ can then be used in (23) to derive $\tau_{\max 2}(k)$. This still neglects the influence the user and the environment will have on the motion of each device. Therefore, it is still possible that more energy is extracted at either side than is stored in the tank. If that were to happen, (21) shuts off the commands from the transparency layer and, thus, prevents the system from becoming unstable.

The two saturation functions mentioned earlier ensure that the system will remain passive or at least minimize the chance of a momentary loss of passivity from occurring. Additional saturation methods can be thought of that not only maintain passivity, but shape the interaction in a beneficial way, which depends on the amount of energy available in the system. An additional saturation method that can be useful is, for instance, to define a mapping $g(H(\overline{k+1}))$ from the current available energy in the tank to the maximum torque that can be applied, which means

$$\tau_{\max 3}(k) = g(H(\overline{k+1})). \tag{24}$$

This mapping can be designed in such a way that a safe interaction in complex situations is guaranteed. What a *safe interaction* is depends on the task, the environment, and the circumstances under which the task has to be executed in the environment. Therefore, no general implementations of (24) can be provided. Two examples, where (24) might be useful, will now be sketched. Assume that an IR algorithm has been implemented in the transparency layer and that the environment has not yet been properly identified. In such a situation, the position controller in the transparency layer at the slave side could exert excessive forces on the environment to track the motions of the master. This is likely to damage the objects that are encountered

in the environment. However, in such situations the amount of energy in the system is very limited (the user is not yet interacting with the virtual model); therefore, (24) could be designed to prevent excessive forces from being applied to the objects in the environment.

A second example, where (24) could be useful, is discussed by Franken *et al.* [33]. There a mapping is used in combination with the SETP to gently release objects, which the slave is grasping when a communication blackout should occur.

The maximum allowable torque $\tau_{\max}(k)$ is the lower bound of all the various limiting/saturation functions:

$$\tau_{\max}(k) = \min(\tau_{\max 1}(k), \tau_{\max 2}(k), \tau_{\max 3}(k), \ldots) \quad (25)$$

where "..." indicate other limiting/saturation functions that can be implemented. These additional functions for instance could be beneficial for a specific device, environment, and/or task to be executed. Note that all limiting functions except $\tau_{\max 1}$ are optional, although the exclusion of $\tau_{\max 2}$ and/or $\tau_{\max 3}$ can result in unwanted switching behavior of the passivity layer. It should also be noted that the limiting/saturation functions in the passivity layer at the master and slave sides do not necessarily have to be identical.

The torque $\tau_{\mathrm{PL}}(k)$, which is the bounded version of the torque $\tau_{\mathrm{TL}}(k)$, that is requested by the transparency layer is computed as

$$\tau_{\mathrm{PL}}(k) = \mathrm{sgn}(\tau_{\mathrm{TL}}(k)) \min(|\tau_{\mathrm{TL}}(k)|, \tau_{\max}(k)). \quad (26)$$

The final torques to be applied to the master and slave devices during the sample periods $\overline{k+1}$, $\tau_{rm}(\overline{k+1})$, and $\tau_{rs}(\overline{k+1})$, respectively, are

$$\begin{aligned} \tau_{rm}(\overline{k+1}) &= \tau_{\mathrm{PL}m}(k) + \tau_{\mathrm{TLC}}(k) \\ \tau_{rs}(\overline{k+1}) &= \tau_{\mathrm{PL}s}(k) \end{aligned} \quad (27)$$

where $\tau_{\mathrm{PL}m}(k)$ and $\tau_{\mathrm{PL}s}(k)$ are the torques computed by the passivity layer at the master and slave sides, respectively. At the master side, $\tau_{\mathrm{TLC}}(k)$ that results from the TLC of (14) is superimposed on $\tau_{\mathrm{PL}m}(k)$ before application to the device.

V. Experimental Results

In this section, we will provide experimental results that were obtained with the setup depicted in Fig. 7. The setup consists of two identical one-degree-of-freedom (1-DOF) devices powered by a dc motor without a gearbox. A high-precision encoder with 65 000 pulses per rotation is used to record the position of each device. The mechanical arm of each device contains a linear force sensor to record the interaction force between the user/environment and the devices. Both devices are controlled from the same embedded controller running a real-time Linux distribution. The controllers are implemented in the program 20-sim [34], and the real-time executable code that is specific for this setup is generated directly from 20-sim and uploaded to the embedded controller by means of the program 4C [34]. The sampling frequency of the control loop is 1 kHz. As an environment, a mechanical spring with a stiffness of approximately 1500 N/m is used. The recorded position of this spring in the environment varies slightly between experiments as incremental position encoders are used, and the initial position of the slave device is not perfectly equal for each experiment.

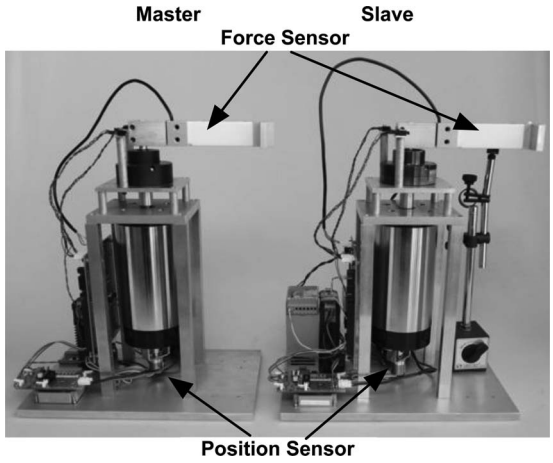

Fig. 7. Experimental setup. The setup consists of two identical 1-DOF devices powered by an electromotor without a gearbox. The position of each motor is recorded with a high-precision incremental encoder, and the mechanical arm consists of a linear force sensor to record the interaction force between the user/environment and the devices.

Two different bilateral controllers were implemented to show the benefits and flexibility of the two-layer approach. A regular position–force (PF) controller is implemented in a situation without and with time delay. In the second experiment, the time delay is still present, and the PF controller is replaced with an IR algorithm. The time delay implemented in the artificial communication channel between the master and slave devices is 1 s, constant, and without package loss. Both experiments are carried out by the use of the same implementation of the passivity layer. Experimental results are shown both with the passivity layer turned ON and OFF and for grasps of the user switching between hard, relaxed, and soft.

First, the implementation of the passivity layer will be discussed and, afterward, the implementations of the two different controllers combined with the experimental results. It should be noted that with the change from the PF controller to the IR algorithm, only the implementation of the transparency layer changes, and no adjustments to the passivity layer are required. There are even no changes needed to the passivity layer when introducing the time delay in the communication channel.

The force sensors record the force at the interaction points between the user/environment and the device. Therefore, it is chosen to have the transparency layer compute a force to be exerted at the interaction point and not directly a torque to be applied by the motor. In the passivity layer, the saturation functions are applied to this force, after which, it is transformed into a torque to be applied by the actuator by the use of the length of the mechanical arm r of each device. For both devices, $r = 0.15$ m. The TLC in the passivity layer at the master side also computes a torque.

A. Implementation Passivity Layer

The passivity layer is implemented as discussed in Section IV. As an energy transfer protocol, the SETP of Section IV-C1

TABLE I
PARAMETER VALUES OF THE PASSIVITY LAYER

Parameter	Value	Parameter	Value
H_D	0.075 J	β	0.01
α	$70\,^{\mathrm{Nm\cdot s}}/_{\mathrm{rad\cdot J}}$	K_S	$500\,^{\mathrm{N}^2}/_{\mathrm{J}}$

TABLE II
PARAMETER VALUES OF THE TRANSPARENCY LAYER

Parameter	Value	Parameter	Value
K_p	$25\,^{\mathrm{N}}/_{\mathrm{rad}}$	K_d	$0.75\,^{\mathrm{Ns}}/_{\mathrm{rad}}$
γ	0.01	β_e	0.99

is chosen. The values for the various parameters are listed in Table I. As saturation functions, (21) and (24) and the maximum force that can be delivered by the actuators have been implemented. The mapping of (24) is implemented only at the slave side and in the form of a linear spring with stiffness K_s; therefore

$$|F_{\max 2}(k)| = \sqrt{2H_S(\overline{k+1})K_S} \tag{28}$$

and

$$|F_{\max 3}(k)| = 11.5\,\mathrm{N}. \tag{29}$$

B. Position–Force Controller

As the first implementation of the transparency layer, a regular PF controller is implemented. Such a controller is characterized by a poor transparency as the proportional gain of the position controller acts as a spring with limited stiffness between the master and slave positions. Although accurate force reflection can be achieved, the position-tracking performance of the slave device will be limited during contact phases with the environment. However, the added benefit of the passivity layer with respect to the stability of the system can clearly be demonstrated. The PF controller is implemented as

$$\begin{aligned} F_{\mathrm{TL}m}(k) &= F_e(k) \\ F_{\mathrm{TL}s}(k) &= -K_p(q_m(k) - q_s(k)) - K_d\dot{q}_s(k) \end{aligned} \tag{30}$$

where F_e is the measured interaction force between the slave device and the environment. K_p and K_d are the gains of the position controller.

Several experiments have been carried out with the controller settings as given in Table II. Each figure shows the position of the master and slave devices, the force recorded by the force sensors in the master and slave devices, the difference between the force computed by the PF controller and the force applied by the passivity layer at the master side (F_{dif}), and the level of the energy tanks at the master and slave sides, respectively.

Fig. 8 shows the response when the passivity layer is not active. As long as the user has a strong grasp on the device, the response is stable, and the interaction forces are accurately reflected. However, when the user relaxes, his grasp oscillations start to occur in the system response when contact is made with the environment. Finally, for a soft grasp, the contact is unstable and the slave system is bouncing on the environment.

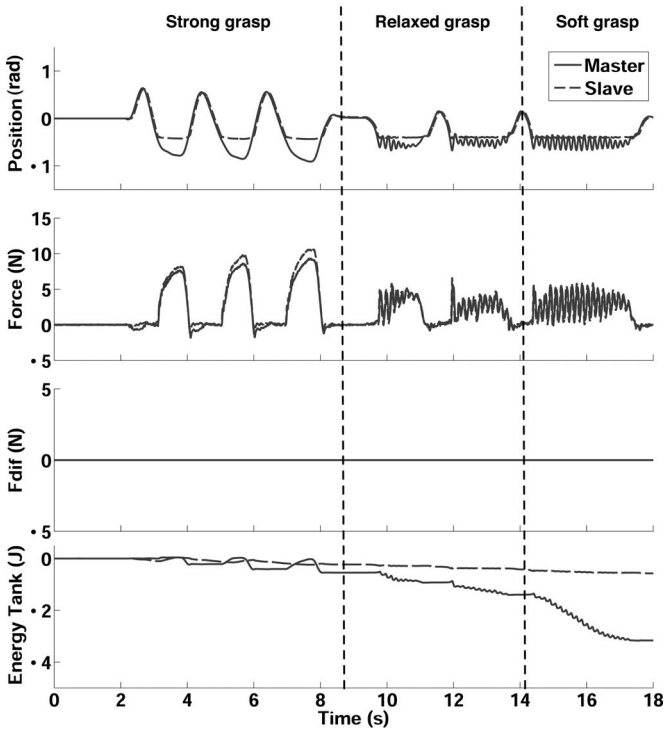

Fig. 8. Nonpassive PF controller. Contact between the slave device and the environment becomes unstable for more relaxed grasps by the user.

The system is producing "virtual" energy as indicated by the negative and decreasing tank level of both the master and slave systems. The energy tank levels also show that in this situation "virtual" energy is mostly generated at the master side as the level of that tank is decreasing much faster than the level of at the slave side.

Fig. 9 shows the same experiment but with the passivity layer activated. The initial extraction phase is indicated in which both energy tanks are filled. For all three grasps, the system response is stable, and the user is experiencing the force feedback. The relative influence of the passivity layer on the feedback force to the user is increasing with more relaxed grasps by the user.

Fig. 10 shows the response of the system when a 1 s delay is introduced in the communication channel. In this case, the transparency of the system is extremely low as action and reaction at the user side are separated in time by a round trip delay of 2 s. Without the passivity layer, a strong grasp is needed in order to keep the system stable as violent jerks frequently occur. Fig. 10 shows that with the passivity layer activated, the system remains stable, even with a relaxed grasp by the user.

C. Impedance Reflection

As the second implementation of the passivity layer, an IR algorithm has been implemented based on the scheme proposed by Tzafestas *et al.* [22], similar to the one used by Franken *et al.* [33]. A schematic drawing of this implementation of the transparency layer is depicted in Fig. 11. The implementation of each element of Fig. 11 is discussed as follows.

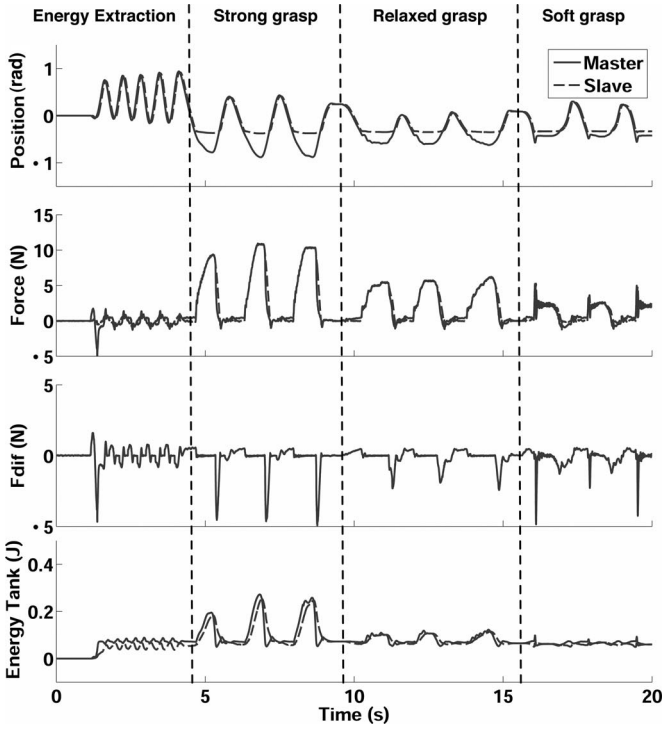

Fig. 9. Passive PF controller. Contact between the slave device and the environment remains stable for all grasps by the user. The relative adjustment by the passivity layer is increasing for more relaxed grasps.

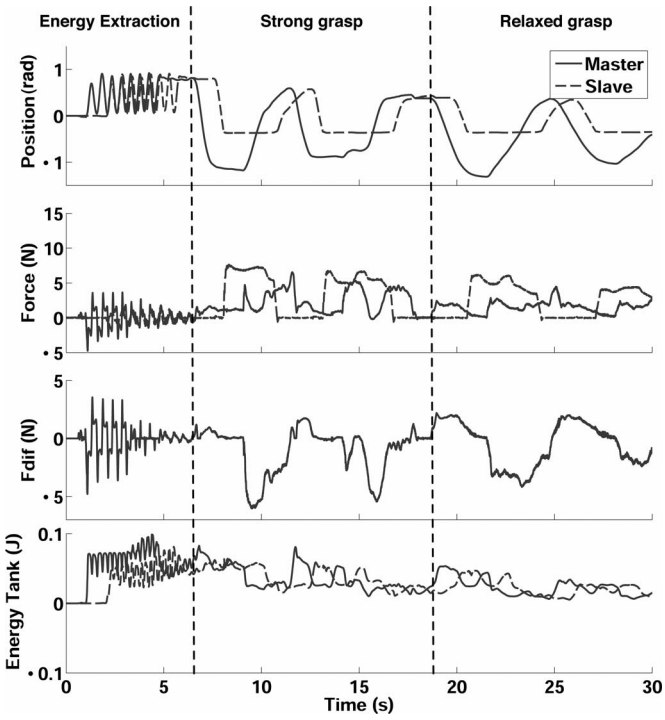

Fig. 10. Passive PF controller with 1 s time delay. Transparency of the telemanipulation system is extremely low due to the time delay but remains stable for even relaxed grasps by the user. Without the passivity layer, violent jerks appear in the feedback force, which require a strong grasp by the user to keep the system stable.

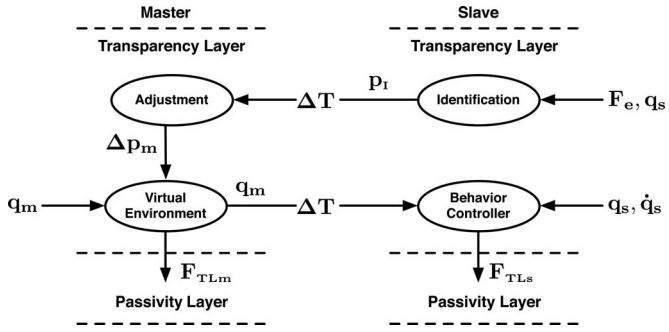

Fig. 11. Impedance reflection algorithm based on [22]. Feedback force to the user is based on a local model of the remote environment of which the parameters are estimated online.

1) Virtual Environment: The virtual environment is implemented as a simple discretized linear spring model:

$$F_{\mathrm{TL}m}(k) = -\hat{K}_e(k)(q_m(k) - \hat{x}_e(k))/r \tag{31}$$

where $\hat{K}_e(k)$ and $\hat{x}_e(k)$ are the estimated stiffness and position of the object in the environment. Due to the identification algorithm, the mechanical spring in the environment is identified as a torsional spring. Therefore, the length of the mechanical arm is used to transform the resulting torque into the desired force at the interaction point.

2) Adjustment: A simple smoothing function is implemented that limits the change in parameters to a percentage of the difference between the currently used and identified parameters:

$$\Delta p_m(k) = \gamma(p_I(k) - p_m(k-1)) \tag{32}$$

where p_I and p_m indicate the received identified parameters and the current parameters used at the master side, respectively.

3) Behavior Controller: The same position controller for the slave device as used in Section V-B is implemented. Therefore

$$F_{\mathrm{TL}s}(k) = -K_p(q_m(k) - q_s(k)) - K_d\dot{q}_s(k). \tag{33}$$

The series spring of the position controller is removed from the feedback force to the user but is still present in the position response of the slave device. This means that the transparency of the system will still be limited as the position responses of the master and slave devices can greatly differ when interacting with the environment. A solution to this problem could be to implement either an adaptive or robust control structure. Misra *et al.* [35], for instance, use the identified parameters of the environment to modify the position control gains.

4) Identification: The identification algorithm implemented here is a linear regression algorithm based on [36]. The estimator tries to minimize the cost function:

$$V_N(\hat{K}_e) = \frac{1}{N}\sum_{k=1}^{N}\epsilon(k)^2$$

$$\epsilon(k) = rF_e(k) - (\hat{K}(k)(q_s(k) - \hat{x}_e) \tag{34}$$

and is implemented by the computation of the following recursive equations during the estimation process:

$$\begin{aligned}\hat{K}_e(k) &= \hat{K}_e(k-1) + Q(k)\epsilon(k)\\ Q(k) &= R(k-1)(\hat{x}_s(k))(\beta_e + (\hat{x}_s(k))^2 R(k-1))^{-1}\\ R(k) &= \frac{1}{\beta_e}(1 - Q(k)\hat{x}_s(k))R(k-1)\\ \hat{x}_s(k) &= q_s(k) - \hat{x}_e(k)\end{aligned} \tag{35}$$

where β_e is a forgetting factor to limit the estimation to more recent measurements. *R* is initialized as 1, and no prior information about the parameters is assumed. $\hat{x}_e$ is determined as the position of the slave device, where a certain force threshold is exceeded. This is done to prevent the dynamics of the force sensor with the slave device that moves in free space to activate the activation algorithm. $\hat{x}_e$ is, therefore, a rough approximation of x_e. This results in an unwanted effect that the estimated stiffness drops to zero when the environment force decreases. Therefore, (35) is only executed when the environment force is increasing. This rough approximation of x_e can also result in the stiffness of the environment to be overestimated.

5) Experimental Results: This scheme improves the transparency of the system with respect to the PF controller in the presence of time delays in the communication channel as the feedback force to the user is predicted based on a local virtual model of the environment. However, the transparency is still limited due to the fixed position controller, as indicated in Section V-C3 and visualized in Figs. 12 and 13 by the large position difference between the master and slave devices.

The settings for the IR algorithm are listed in Table II. The time delay in the communication channel is again 1 s. Each figure shows the position of the master and slave devices, the force recorded by the force sensor in each device, the estimated stiffness by the identification algorithm of Section V-C4 expressed as radial stiffness, and the level of the energy tanks at the master and slave sides, respectively.

Fig. 12 shows the system response when the passivity layer is not activated. While the slave device interacts with the environment, the identification algorithm estimates the stiffness of the environment. The user is subsequently presented with a predictive force feedback that is based on the implemented model and identified parameters. The initial force exerted by the slave device on the environment is 12 N. This force is only limited due to the limitations of the motor amplifiers and would have been much higher. It should be noted that the force computed by the local model, for the identified stiffness, in the transparency layer is saturating the motor amplifiers. This means that the environment can only be accurately reflected within a certain position range of the master device. When the user relaxes his grasp on the device, the interaction with the local virtual model is no longer stable, and large oscillations in the feedback force occur. This means that the system is generating energy, as is visible from the negative and rapidly decreasing tank level of the master device. This oscillatory behavior is, subsequently, also exhibited by the slave device.

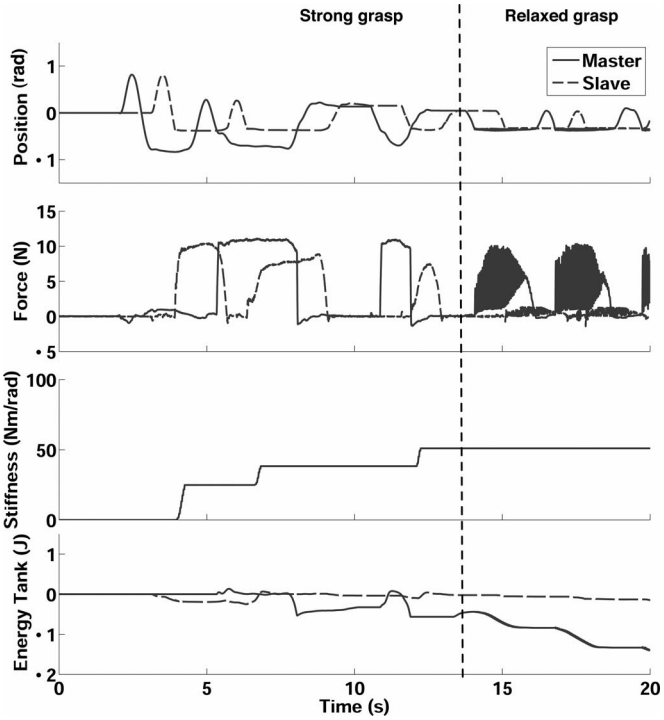

Fig. 12. Nonpassive IR. Interaction force is now predicted based on the local model, which increases the transparency of the system with respect to the PF controller. The initial contact between the slave and the environment is a very hard collision and only limited due to the saturation of the motor amplifiers. For a relaxed grasp of the user, the contact with the virtual model is unstable.

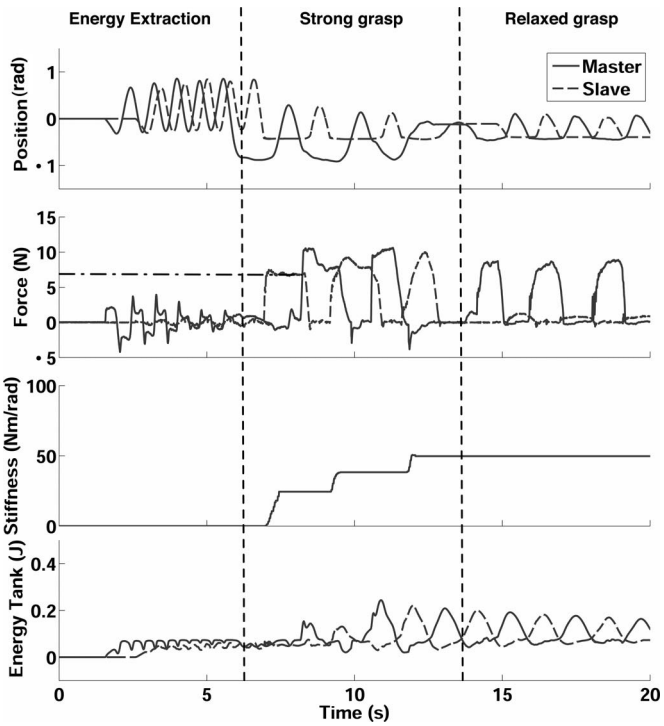

Fig. 13. Passive IR. Increase in transparency with respect to the PF controller remains. With respect to the nonpassive implementation, the initial contact can be shaped by means of the saturation functions in the passivity layer. Due to the added damping by the passivity layer, the contact with the virtual model is stable also for a relaxed grasp of the user.

Fig. 13 shows the response with the passivity layer activated. Two major differences are visible with respect to Fig. 12. The initial impact force between the slave device and the environment is limited due to the saturation functions in the passivity layer. The impact force in Fig. 13 is about 7 N, whereas it is 12 N in Fig. 12. When the user is interacting with the properly identified local model, he/she is injecting energy into the system, which allows the passivity layer at the slave system to exert higher forces on the remote environment as the user clearly intends to exert these forces on the object. The second difference is the absence of the vibrations when the user is relaxing his grasp. The interaction with the virtual environment is kept stable due to the added damping in the passivity layer to keep the system passive.

It should be noted that there exists a lower bound of the user's grasp for this latter effect. The passivity layer only adds enough damping to maintain passivity of the system. As the virtual environment is a pure undamped spring, it will exhibit an oscillatory response when there is no damping added by the user's grasp. This oscillatory behavior is simply stable, and it does not grow in magnitude, as the passivity layer will enforce passivity of the system. In the absence of the passivity layer, the system will be unstable for the same soft grasp by the user.

VI. Discussion

With respect to each of the passivity-based control structures that are treated in Section II, the proposed two-layer framework with the implementation of the passivity layer treated in Section IV has at least some of the following benefits.

1) *Hardware independent:* No additional relation between the implementation of the controller and the hardware parameters is needed to ensure stable behavior.
2) *Wide variety of bilateral controllers:* The only restriction placed on the implemented bilateral controller is that it computes a force to be applied to each device.
3) *Almost independent optimization of each layer:* The manner in which the passivity layer is implemented monitors the energy exchange and only intervenes when necessary. This means that if the bilateral controller in the transparency layer is displaying passive behavior, the passivity layer does nothing. This allows almost independent optimization of each layer. The only dependence between the layers is that the energy transfer protocols that can be implemented are restricted by the chosen implementation of the transparency layer.
4) *Flexibility:* The passivity layer is centered around the communicating energy tanks and the monitoring of the energy exchange with the physical world. Any number of saturation functions can be designed, implemented, and optimized independently of each other to shape that physical interaction based on the available energy in the tank.

As has been pointed out in Section IV, it is possible for the system to be momentarily active. This is inherent to the fact that the energy exchange during the sample period cannot be monitored, and its value can only be computed *a posteriori*. All time domain passivity structures that are centered around monitoring the energy exchange share this effect. However, in this framework, the active behavior of the system is limited to a single sample period, because the passivity layer will shut off the commands of the transparency layer until passivity of the system is restored. Other approaches such as the EBA and PSPM can prevent energy from being generated at all times but only as long as the models of the hardware that are being used are accurate lower bounds.

VII. Conclusion and Future Work

In this paper, a new framework for bilateral telemanipulation was presented. The two-layer approach allows the combination of passivity and transparency in a very intuitive manner. By the use of this framework, any control architecture with an impedance causality can be implemented in a passive manner. Furthermore, the framework allows many of its features to be tuned for specific devices and/or tasks. Especially, the energy transfer protocol and saturation functions can be designed and optimized for a specific device, environment, and/or task. The presented experimental results show the benefits of the two-layer implementation. A single implementation of the passivity layer was able to maintain stability of two different implementations of the transparency layer, even in the presence of large time delays, hard contacts, and a variety of user grasps. The transparency properties of the bilateral controllers, which were implemented in the transparency layer, were maintained, and their commands were only adjusted by the minimum to maintain passivity of the system.

Future work will focus on the systematic implementation of the various design options and tuning of the parameters in the passivity layer. In addition, the implementation of the framework on systems with multiple degrees of freedom will be analyzed.

The passivity layer presented in this paper makes the system passive with respect to the actuators at both the master and slave sides. All the energy spent by the actuators at the slave side is extracted from the user. This means that transparency is adversely influenced by friction in the slave device. Therefore, future research will also be directed to friction-compensation techniques to extend this approach to manipulators with high internal friction. Preliminary results of such a friction-compensation technique were presented by Franken *et al.* [37].

References

[1] B. T. Bethea, A. M. Okamura, M. Kitagawa, T. P. Fitton, S. M. Cattaneo, V. L. Gott, W. A. Baumgartner, and D. D. Yuh, "Application of haptic feedback to robotic surgery," *J. Laparoendoscopic Adv. Surg. Tech.*, vol. 14, no. 3, pp. 191–195, 2004.
[2] D. Lawrence, "Stability and transparency in bilateral teleoperation," *IEEE Trans. Robot. Autom.*, vol. 9, no. 5, pp. 624–637, Oct. 1993.
[3] T. B. Sheridan, "Telerobotics," *Automatica*, vol. 25, no. 4, pp. 487–507, 1989.
[4] T. B. Sheridan, "Space teleoperation through time delay: Review and prognosis," *IEEE Trans. Robot. Autom.*, vol. 9, no. 5, pp. 592–606, Oct. 1993.
[5] P. Hokayem and M. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, pp. 2035–2057, 2006.
[6] A. van der Schaft, *L2-Gain and Passivity in Nonlinear Control*. New York: Springer-Verlag, 1999.

[7] N. Hogan, "Controlling impedance at the man/machine interface," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1989, pp. 1626–1631.
[8] R. Anderson and M. Spong, "Bilateral control of teleoperators with time delay," *IEEE Trans. Autom. Control*, vol. 34, no. 5, pp. 494–501, May 1989.
[9] G. Niemeyer and J.-J. E. Slotine, "Telemanipulation with time delays," *Int. J. Robot. Res.*, vol. 23, no. 9, pp. 873–890, 2004.
[10] P. Arcara and C. Melchiorri, "Control schemes for teleoperation with time delay: A comparative study," *Robot. Auton. Syst.*, vol. 38, no. 1, pp. 49–64, 2002.
[11] C. Lawn and B. Hannaford, "Performance testing of passive communication and control in teleoperation with time delay," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1993, pp. 776–783.
[12] C. Secchi, S. Stramigioli, and C. Fantuzzi, *Control of Interactive Robotic Interface* (Springer Tracts in Advanced Robotics), vol. 29. New York: Springer-Verlag, 2006.
[13] G. Niemeyer, "Using wave variables in time delayed force reflecting teleoperation," Ph.D. dissertation, Dept. Aeronaut. Astronaut., MIT, Cambridge, MA, 1996.
[14] H. Ching and W. J. Book, "Internet-based bilateral teleoperation based on wave variable with adaptive predictor and direct drift control," *Trans. ASME, J. Dyn. Syst., Meas., Control*, vol. 128, no. 1, pp. 86–93, 2006.
[15] N. Tanner and G. Niemeyer, "Improving perception in time-delayed telerobotics," *Int. J. Robot. Res.*, vol. 24, no. 8, pp. 631–644, 2005.
[16] J.-H. Ryu, D.-S. Kwon, and B. Hannaford, "Stable teleoperation with time-domain passivity control," *IEEE Trans. Robot. Autom.*, vol. 20, no. 2, pp. 365–373, Apr. 2004.
[17] B. Hannaford and J.-H. Ryu, "Time domain passivity control of haptic interfaces," *IEEE Trans. Robot. Autom.*, vol. 18, no. 1, pp. 1–10, Feb. 2002.
[18] J. Artigas, C. Preusche, and G. Hirzinger, "Time domain passivity for delayed haptic telepresence with energy reference," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2007, pp. 1612–1617.
[19] J. Artigas, C. Preusche, G. Hirzinger, G. Borghesan, and C. Melchiorri, "Bilateral energy transfer in delayed teleoperation on the time domain," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 671–676.
[20] J.-H. Ryu and C. Preusche, "Stable bilateral control of teleoperators under time-varying communication delay: Time domain passivity approach," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2007, pp. 3508–3513.
[21] J.-H. Ryu, J. Artigas, and C. Preusche, "A passive bilateral control scheme for a teleoperator with time-varying communication delay," *Mechatron.*, vol. 20, pp. 812–823, 2010.
[22] C. Tzafestas, S. Velanas, and G. Fakiridis, "Adaptive impedance control in haptic teleoperation to improve transparency under time-delay," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 212–219.
[23] K. Kawashima, K. Tadano, G. Sankaranarayanan, and B. Hannaford, "Model-based passivity control for bilateral teleoperation of a surgical robot with time delay," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2008, pp. 1427–1432.
[24] J.-P. Kim and J. Ryu, "Robustly stable haptic interaction control using an energy-bounding algorithm," *Int. J. Robot. Res.*, vol. 29, no. 6, pp. 666–679, 2010.
[25] C. Seo, J. Kim, J.-P. Kim, J. H. Yoon, and J. Ryu, "Stable bilateral teleoperation using the energy-bounding algorithm: Basic idea and feasibility tests," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatron.*, 2008, pp. 335–340.
[26] D. Lee and K. Huang, "Passive-set-position-modulation framework for interactive robotic systems," *IEEE Trans. Robot.*, vol. 26, no. 2, pp. 354–369, Apr. 2010.
[27] D. J. Lee and K. Huang, "Passive position feedback over packet-switching communication network with varying-delay and packet-loss," in *Proc. Symp. Haptic Interf. Virtual Environ. Teleoperator Syst.*, 2008, pp. 335–342.
[28] J. Colgate, P. Grafting, M. Stanley, and G. Schenkel, "Implementation of stiff virtual walls in force reflecting interfaces," in *Proc. IEEE Virtual Reality Annu. Int. Symp.*, 1993, pp. 202–208.
[29] J. Abbott, P. Marayong, and A. M. Okamura, "Haptic virtual fixtures for robot-assisted manipulation," in *Proc. 12th Int. Symp. Robot. Res.*, 2007, vol. 28, pp. 49–64.
[30] S. Stramigioli, C. Secchi, A. van der Schaft, and C. Fantuzzi, "Sampled data systems passivity and discrete port-Hamiltonian systems," *IEEE Trans. Robot. Autom.*, vol. 21, no. 4, pp. 574–587, Aug. 2005.
[31] Z. Ren, H. Zhang, and H. Shao, "Robust stability analysis of discrete-time linear systems with time delay," in *Proc. Am. Control Conf.*, 2003, pp. 4840–4844.
[32] E. Jonckheere and C. Ma, "A further simplification to Jury's stability test," *IEEE Trans. Circuits Syst.*, vol. 36, no. 3, pp. 463–464, Mar. 1989.
[33] M. Franken, S. Stramigioli, R. Reilink, C. Secchi, and A. Macchelli, "Bridging the gap between passivity and transparency," in *Proc. Robot.: Sci. Syst.*, Jun. 2009, pp. 281–288.
[34] Controllab Products B.V. (2010). 20-sim ver. 4.1 [Online]. Available at http://www.20sim.com/
[35] S. Misra and A. M. Okamura, "Environment parameter estimation during bilateral telemanipulation," *Proc. Symp. Haptic Interf. Virtual Environ. Teleoperator Syst.,* 2006, pp. 301–307.
[36] N. Diolaiti, C. Melchiorri, and S. Stramigioli, "Contact impedance estimation for robotic systems," *IEEE Trans. Robot.*, vol. 21, no. 5, pp. 925–935, Oct. 2005.
[37] M. Franken, S. Misra, and S. Stramigioli, "Friction compensation in energy-based bilateral telemanipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 5264–5269.

Michel Franken (S'08) received the B.Sc and M.Sc. (Hons.) degrees in electrical engineering with a focus on measurement and control engineering from the University of Twente, Enschede, the Netherlands, in 2004 and 2007, respectively. His M.Sc. thesis was on ankle actuation of planar bipedal robots under the supervision of Prof. S. Stramigioli. He is currently working toward the Ph.D. degree from the University of Twente with the Control Engineering group of Prof. S. Stramigioli in the area of control methods for haptic feedback and bilateral telemanipulation.

He was a Visiting Research Scholar with Oakland University, Rochester, MI, in 2006, where he worked on autonomous navigation of mobile robots in collaboration with JADI, Inc.

Mr. Franken's M.Sc. project was recognized with the Mechatronic Valley Twente Award for the Best Mechatronics Graduation Project in 2008.

Stefano Stramigioli (S'96–M'98–SM'03) received the M.Sc. degree (*cum laude*) in electrical engineering with specialization in control systems from the University of Bologna, Bologna, Italy, and the Ph.D (*cum laude*) degree in electrical engineering from the Delft University of Technology, Delft, The Netherlands, 1992 and 1998, respectively.

He was a researcher with the University of Twente in between his M.Sc. and Ph.D. projects. Since 1998, he has been a Faculty Member and is currently a Full Professor of Advanced Robotics and the Chair Holder of the Control Engineering Group, University of Twente. He is an Officer of the IEEE. He has been the Director of the Strategic Research Orientation of Institute of Mechanics, Process, and Control, Twente, University of Twente. He has more than 150 publications, including four books, book chapters, and journal and conference contributions.

Dr. Stramigioli is the Emeritus Editor in Chief of the IEEE ROBOTICS AND AUTOMATION MAGAZINE, which became the Robotics journal with the highest IF (3.0), and the Emeritus Editor in Chief of the *IEEE ITSC Newsletter*. He is a member of the Editorial Board of the *Springer Journal of Intelligent Service Robotics* and is currently the Vice President for Member Activities of the IEEE Robotics and Automation Society (IEEE RAS). He has been an AdCom member for IEEE RAS. He is a member of the European Space Agency Topical Team on Dynamics of Prehension in Microgravity and its application to Robotics and Prosthetics, Chair of RoboNED, which is the Dutch Robotic Network, and cofounder of the LEO Center for Service Robotics.

Sarthak Misra (S'05–M'10) received the M.Sc. degree in mechanical engineering with a focus on multi-body dynamics of space systems with reaction wheels from McGill University, Montreal, QC, Canada, in 2001 and the Ph.D. degree in mechanical engineering with a focus on realistic tool-tissue interaction modeling from the Johns Hopkins University, Baltimore, MD, in 2009.

He joined the University of Twente, Enschede, the Netherlands, as an Assistant Professor in 2009. He is a member of the Control Engineering Group and affiliated with the Institute for Biomedical Technology and Technical Medicine, University of Twente. Prior to commencing his studies at Johns Hopkins, he worked for three years as a Dynamics and Controls Analyst on the International Space Station Program. His research interests include biomechanics and robotics.

Dr. Misra is the recipient of a 2010 Netherlands Organization for Scientific Research VENI Award. He is an Associate Editor of the IEEE Robotics and Automation Society Conference Editorial Board.

Cristian Secchi (S'01–M'04) graduated in Computer Science Engineering, with majors in automation and robotics, from the University of Bologna, Bologna, Italy, in 2000 and received the Ph.D. in information engineering, curriculum automation, from the University of Modena and Reggio Emilia, Reggio Emilia, Italy.

He is currently an Assistant Professor with the University of Modena and Reggio Emilia. He is a coauthor of the book *Control of Interactive Robotic Interfaces: A Port-Hamiltonian Approach* (Berlin, Germany: Springer, 2007). His main research interests include teleoperation, mobile robotics, and the control of mechatronic systems.

Dr. Secchi was selected as one of the three finalists for the Georges Giralt Award for the best Ph.D. dissertation on robotics in Europe in 2006. He participated in the Coordination of AGVs in Automatic Warehouses Project, which was selected as one of the finalists for the 2010 EUROP/EURON Technology Transfer Award. He has been an Associate Editor of the *IEEE Robotics and Automation Magazine* from 2006 to 2008 and the Co-Chair of the IEEE Robotics and Automation Society Technical Committee on telerobotics since 2007.

Alessandro Macchelli (M'06) received the M.Sc. degree (*cum laude*) in computer science engineering in 2000 and the Ph.D. degree in 2003, both from the University of Bologna, Bologna, Italy.

In 2001, he was appointed as a Visiting Scholar and, again, in 2003, got a Postdoctorate position with the Department of Applied Mathematics, University of Twente, Enschede, the Netherlands. Since 2005, he has been a Faculty Member with the Department of Electronics, Computer Science, and Systems, University of Bologna, as an Assistant Professor. His research interests include the modeling, simulation, and control aspects of finite- and infinite-dimensional systems within the port-Hamiltonian framework.