BILATERAL TELEOPERATION OF MULTIPLE COOPERATIVE ROBOTS FOR LOAD TRANSPORT

GABRIEL C. Motta-Ribeiro, FERNANDO LIZARRALDE and LIU HSU*

*Dept. of Electrical Engineering, COPPE/Federal University of Rio Janeiro, Rio de Janeiro, Brazil.

Email: [fernando, liu]@coep.ufrj.br

Abstract— In this paper the bilateral teleoperation of multiple cooperative robots is addressed. A control framework for the teleoperation between multiple master/slave robots is proposed for collaboratively perform a task. A recently proposed two-layer approach where the control architecture is separated into two layers is considered. The top layer is used to address the transparency and the lower layer guarantees that no virtual energy is generated. As a cooperative task, the load transportation problem is considered which is treated as a formation control problem where the contact force between robots and the payload are modeled as gradients of nonlinear potentials. A decentralized ultimately bounded control scheme is proposed for controlling each one of the robot manipulators. Without explicit communication between the robots, all robot manipulators and the payload track a velocity reference with bounded contact forces. Numerical simulation and experimental results using a Phantom Omni robot and a Motoman dual-arm robot illustrate the feasibility of the proposed approach.

Keywords— Teleoperation, Load Transport, Cooperative Robotics

1 Introduction

Haptic teleoperation of robotic systems has great potentials in a wide variety of tasks because it allows an operator, in a safe environment, to remotely control a robot located in a distant and/or dangerous place. Haptic and visual feedback give to the operator a notion of telepresence which greatly enhances the performance of the operation. Stability and transparency of teleoperated systems have been studied in detail in literature using for examples passivity-based [11, 8, 14] and energy-based approaches [5, 18].

An interesting application of teleoperated systems is remote load handling of large objects. Because handling large objects requires two or more robots to work as a team, additional challenges that are not present in standard teleoperated systems arise. As one example, as common the telepresence experienced by the operator is never perfect, mainly due to communication delays, data package loss, and limited visual feedback, and as a result the force closure cannot be left solely for the operator. To guarantee force closure a safety layer on the slave side needs to be implemented [4].

This paper is concerned with the problem of teleoperated multiple robotic manipulators to jointly transport a load. First, we need to guarantee that the closed kinematic chain is maintained at all times and that sufficiently high grasping forces are applied to hold the object. At the same time we want to maintain the possibility of the operator control over the interaction forces. Secondly, we need to guarantee stability of the bilateral system. This becomes increasingly difficult when the number of master and slave manipulators grows large. Finally, we need to maintain the transparency of the system in the presence of the constraints imposed on the slave side. This means that if 2 master robots control one slave robot each, and a constraint is imposed on the slave robots to maintain the kinematic chain, this constraint also needs to be imposed on the master.

Force control issues and cooperative control in teleoperated systems have also been discussed in literature [15, 17, 2, 12, 1]. In [12, 1] a single master commands a object rigidly grasped by multiple robots. Considering a multimaster/multislave architecture [17] proposes a controller such that position and force information are shared between all agents, while master and slave are paired. Using the concept of paired remote and local robots, [2] suggest the use of multiple individual teleoperators acting on a shared environment, with the same tool.

On a variety of solutions to the problem of a group of robots collaboratively transporting a payload, in [3] an agent formation-control strategy is proposed. The interaction forces build an implicit communication in the group which makes it possible to implement a decentralized control strategy where convergence to the desired load velocity and force regulation are guaranteed. This method does not require any knowledge about the load for the control design and assumes a deformable object (the system complacency is not neglected). Here, only a Cartesian linear robot without orientation control is considered and the system stability is only guarantee for velocity regulation.

In this paper, the load transport strategy extends of [3] in the sense that considers the tracking of a velocity reference with a full kinematic chain manipulator, which lead to the necessity of end-effector orientation control. In particular, the load handling is implemented on a kinematic level which allows implementation on a standard industrial manipulator. For manipulators with non-negligible dynamics a cascade control can be implemented [9]. The task is accomplished using a multimaster/multislave approach, coupling pairs
of robots only for the force feedback. To solve the bilateral teleoperation problem, a recently proposed two-layer approach is considered. The force closure and motion control are implemented in a high-level transparency layer, while the stability is guaranteed by a passivity layer [5, 18].

2 Teleoperation Architecture

Here we propose a novel control architecture well suited for load transportation using teleoperated robots. The load transportation is based on the approach of formation control in [3] as presented in the next section. The slave robots are intended to follow the master robots end-effector velocity as closely as possible subject to the constraints imposed by the force closure. These forces are also communicated to the operator by imposing a force which tends to keep the distance between the master robots constant. Also force feedback from the interaction forces with the object is added. Finally, following the approach in Franken et al. [5], stability is guaranteed by implementing a passivity layer that monitors the energy flow in the system. This represents a dual-layer architecture which can be treated as independent on the design level.

2.1 Transparency Layer

In the setting of this paper we want to allow the operator to squeeze the object, control the trajectory of the load, and to communicate both the constraints imposed on the slave side and the interaction forces to the operator. The N masters velocities are transmitted to the slave side and used to calculate the desired velocity ($v^d$) as

$$v^d = \frac{1}{N} \sum_{i=1}^{N} \dot{x}^m_i$$

where $\dot{x}_i$ is the velocity of the $i^{th}$ master and the superscript $m$ indicates master variables. The desired force ($f^d$) applied to the load by each slave is

$$f^d = f^{min}_i + f^{dm}_i$$

composed by a minimum force $f^{min}_i$ to guarantee the load grasp and a user force $f^{dm}_i$ which permits the operator to apply additional forces to the object. Note that the command relative to $f^{min}_i$ should not be computed on the energy balance (4). The portion relative to the user is obtained from the masters position considering a virtual deformable object modeled as linear springs between the masters’ end-effector and the center of this object represented by $f^{dm}_i$.

The slave and master robots are grouped by the direction, relative to an inertial coordinate system, of the contact force. In that way, there is no direct pairing through the two sides and the number of remote and local devices can be different. That way, the squeeze force reference is obtained by

$$f^{dm}_i = \frac{k_{fm}}{n_i} \sum_{j \in \Omega_d} \left( \|x^m_j - x^m_i\| - L \right) \vec{d}$$

for each group $\Omega_d$ of robots, were $\vec{d}$ is the direction of the contact force with the virtual or real load.

We note that $f^{min}_i$ is dealt on the slave side while $f^{dm}_i$ is the reference received from the master side. The stability of the load, i.e., to maintain a closed kinematic chain with sufficient grasping forces, is handled on the slave side and is not influenced by time delays and package losses, etc. The additional user force $f^{dm}_i$, however, is influenced by time delays and the stability issues that arise when teleoperated robots interact with a flexible environment are thus very much present in this system.

For the force feedback the robots are also grouped by the force direction and only the force error relative to the minimum is considered, i.e.,

$$f^e_i = \frac{1}{n_i} \sum_{j \in \Omega_c} \| f_i \| - \| f^{min}_j \|$$

where $f_i$ are the $i^{th}$ slave end-effector forces.

2.2 Passivity Layer

Once that transparency layer is designed, the passivity layer is implemented to guarantee stability. This layer can be implemented independently of the transparency one. The passivity layer monitors the energy flow in the system and guarantees that no virtual energy is created. As proposed in [5], at each sample instant $k$ the energy available to move the manipulators can be estimated by

$$H_j(k) = H_j(k-1) + H_{j+} - H_{j-} - \Delta H_j$$

with subscript $j$ being $m$ for the master and $s$ for the slave. $H_j(k-1)$ is the energy level at the previous time step and the terms $H_{j+}$ and $H_{j-}$ are used to exchange energy between the two sides. One simple way to exchange energy is to send energy packets consisting of a fraction of the stored energy at each side, i.e., let

$$H_{j-} = \beta H_j(k-1)$$

be transmitted to the other side and stored in a queue ($Q_j$) until the next motion controller sample instant. The additive term is computed from this queue as

$$H_{j+} = \sum_{j' \in Q_j} H_{j'}$$

and the queue is emptied at each sample instant. This approach endeavors to make the energy as
equally distributed as possible so that both the masters and the slaves are allowed to move.

The final term $\Delta H_j$ is the energy exchanged between the robot and the environment/user. This term can either add or remove energy. The energy is easily calculated for a discrete control system using the measured force $f_j$ and the calculated inertial positions $x_j$ by

$$
\Delta H = f_j^T (x_j(k) - x_j(k - 1)).
$$

(7)

We note that with a robot under dynamic control only position measurements are required, considering the commanded force instead of the measured [19].

We need to add energy to the system to guarantee that the system does not die out. This is done on the master side by applying a small additional force ($F_{TLC}$) that opposes the motion of the operator. This force is defined as

$$F_{TLC} = \alpha \cdot (H_d - H_m(k)) \hat{x}_m
$$

and is applied only when the tank is below the desired energy level $H_d$.

Energy storage tanks are implemented on both the master and the slave sides. The stability of the system is guaranteed by a passive transfer of energy between the two tanks and by assuring that the robots only move if there is enough energy present in the corresponding energy tank. This restriction is done using a saturated force control signal $F_j$

$$F_{j}^{sat} = \min(F_{max1}, \|F_j\|) \frac{F_j}{\|F_j\|}
$$

(9)

with $F_{max1}$ being zero when no energy is available. This permits applying other limits useful to specific application, as example restrict a maximum force or release an object smoothly [5]. The saturation guarantee that any virtual energy due to delays at the communication channel of the Transparency Layer do not affect the system stability.

On a kinematic control loop, common to industrial robots, the saturation could simply be done on the velocity, instead of the force. That way (9) will be

$$\dot{x}_j^{sat} = \min(\dot{x}_{max1}, \|\dot{x}_j\|) \frac{\dot{x}_j}{\|\dot{x}_j\|},
$$

(10)

with $\dot{x}_{max1}$ corresponding to $F_{max1}$.

3 Cooperation control

Consider $N$ robots holding a deformable load with frictional point contact, such that only forces and not moments are transmitted. Let us suppose that the load is initially undeformed, and that robot $i$ is attached to the load at point $a_i$. With the load orientation given by the rotation matrix $R(\theta_c)$ one has:

$$a_i(t) = x_i(t) + R(\theta_c)r_i$$

and

$$\dot{a}_i(t) = \dot{x}_i(t) + \dot{\theta}_c R(\theta_c)r_i$$

where $x_i$ is the inertial position of the load center of mass, $x_i$ is the inertial position of the robot $i$ end-effector, and $r_i$ is a fixed vector in the inertial frame.

The position $a_i(t)$ represents where robot $i$ would be attached as if the load were undeformed at time $t$. As robots move, the flexible load could be squeezed since $x_i \neq a_i$, defining a deformation $z_i = x_i - a_i$ which generates a reaction force $f_i$ to robot $i$.

This force could be seen as the gradient ($\nabla P(z_i)$) of a positive-definite potential function of deformation. As stated above when $z_i = 0$ the robot $i$ isn’t deforming the object, so no reaction force is generated and the condition

$$P(z_i) = 0 \iff z_i = 0
$$

(11)

$$\nabla P(z_i) = 0 \iff z_i = 0
$$

(12)

must be satisfied by $P(z_i)$ [3]. Note that as $a_i$ lies on the undeformed object external face, $z_i$ and then $f_i$ are actually limited at 0. This represents a non pull condition of the end-effector contact.

The load dynamics, which are restricted to purely translation motion, are modeled by

$$M_c \ddot{x}_c = \sum_{i=1}^{N} f_i
$$

(13)

where $M_c$ is the mass of the load and $f_i$ ($i = 1, ..., N$) the contact forces of the robots end-effectors with the object. The total system equations, including the robots, are given as a $N+1$ cooperative system modeled according the relevance of slaves dynamics.

Because the load is eventually moving with a constant velocity, the robots are subject to the following constraint:

$$\sum_{i=1}^{N} f_i^d = 0.
$$

(14)

The setpoints $f_i^d$ should also be chosen so that the contact forces $f_i$ have the desired properties, such as force closure. This requires the knowledge of the payload geometry and the grasping points.

Noting that the reaction force $f_i$ depends on the deformation $z_i$, if $z_i$ can be regulated to some desired state, $f_i$ would also be maintained accordingly. To this end, we assume that $f_i^d$ is modeled by a linear spring-force model. Thus achieving a desired contact force $f_i^d$ is equivalent to driving the deformation $z_i$ to $z_i^d$. This is similar to the formation control problem, where the relative positions between agents are driven to some desired values [3].
3.1 Negligible Dynamics

When the slaves dynamics can be neglected, for example on industrial robots, these are modeled as redundant kinematic chains with n revolute joints. This leads to the differential kinematic model

\[ \dot{x}_i = J_i(\theta_i)u_i \]  

where \( J_i(\theta_i) \) is the portion relative to linear velocity on the Jacobian matrix, \( \theta_i \in \mathbb{R}^n \) are the joints position and \( u_i \approx \dot{\theta}_i \in \mathbb{R}^n \) is a control signal on joint space.

To achieve the goal on move around the load while the grasp is conserved a Cartesian control loop is proposed taking in account the redundancy to perform subtasks. The decentralized cooperative control is given by

\[ u_i = J_i^\dagger \phi_i + J_i^\# \mu_i \]  

where \( J_i^\dagger = J_i^T(J_iJ_i^T)^{-1} \) is the Moore-Penrose pseudo-inverse of the Jacobian and \( J_i^\# = (I - J_iJ_i^\dagger) \) the null space projection matrix. The control signals \( \phi_i \) and \( \mu_i \) are designed to accomplishing the main and secondary task, respectively, i.e. to move the load and regulate end-effector attitude.

3.1.1 Main task

The main task of the cooperative system is to track a desired velocity reference \( v^d(t) \) with the manipulators end-effectors and the load, while this is maintained secure by a bounded force. For this a Cartesian kinematic control loop is proposed as

\[ \phi_i = v^d + K_i (f^d_i - f_i) \]  

where \( K_i \) is the force control gain.

Considering the closed-loop (15)–(17) and defining the velocity errors \( \xi_i(t) = \dot{x}_i(t) - v^d(t) \) and \( \zeta_i(t) = \dot{z}_i(t) - v^d(t) \), the complete Cartesian space closed loop equations are given by

\[ \begin{align*}
\xi_i &= K_i (f^d_i - \nabla P(z_i)) \quad (18) \\
\zeta_i &= \xi_i - \zeta_e \quad (19) \\
\dot{\zeta}_e &= -\dot{v}^d(t) + M_c^{-1} \sum_{i=1}^N \nabla P(z_i) \quad (20)
\end{align*} \]

where we consider the contact force definition (3) and the fact that \( \mu_i \) is on the null space of \( J_i \).

The interactions between the \( N + 1 \) robots then display a star topology with the payload at the center. Therefore, controlling the forces (and, thus, the deformations) between the robots and the payload simultaneously guarantees that the relative positions between the agents are maintained tightly. We also note that the payload is a passive agent that has no direct control input. Recovering the \( \dot{v}^d(t) \) information from the contact forces \( f_i \), which are, indeed, the local interactions with the other agents.

Theorem 1 Consider a cooperative system modeled by (15) transporting a deformable load (13). The interaction forces are considered as the gradient of a positive potential function of the object deformation, submitted to (12). Suppose that the robots are controlled by (16) using (17). If the velocity reference \( v^d(t) \) and its derivative \( \dot{v}^d(t) \) are bounded and the grasp force \( f^d \) satisfy (14), then the complete system (20) practically track (i.e. is ultimate bounded) \( v^d(t) \) with bounded \( f_i \).

Proof: see appendices.

3.1.2 Subtask

The subtask controller uses the joint free-motion on the null space to preserve the orientation of manipulator end-effector perpendicular to the object. So consider a convex cost function \( F(\theta) \) chosen such that its minimum is the correct manipulator orientation. Then the subtask consist in minimize \( F(\theta) \).

To achieve this the proposed null space control signal is

\[ \mu_i = -K_n \frac{\partial F(\theta_i)}{\partial \theta_i} \]  

with \( K_n \) a positive definite gain matrix. This consist in an velocity in the direction of the cost function gradient, so pointing toward the minimum and equal zero on that point.

An example of a suitable cost function is given by

\[ F(\theta_i) = (\psi(\theta_i) - \psi_d)^2 \]  

where \( \psi(\theta_i) \) is a function that describe the manipulator orientation in term of joint angle and \( \psi_d \) is the desired end-effector orientation.

Remark 1 This subtask consist in a local optimization constrained by the main task, the global solution is not guarantee to exist. Especially when the manipulator is on a singular configuration, the are no projection on null space.

The control law (21) is a general solution to minimize a second objective that consist of a convex cost function.

3.2 Nonnegligible Dynamics

Now, considering the control problem for a robot manipulator with nonnegligible dynamics (e.g., direct-drive manipulators), an extension of the proposed controller to include the robot dynamics is presented. The nonlinear dynamic model of the robot manipulator in contact with the environment can be expressed in generalized coordinates

\[ M(\theta) \ddot{\theta} + C(\theta, \dot{\theta}) \dot{\theta} + g(\theta) = Y(\theta, \dot{\theta}, \ddot{\theta}) a = \tau + J^T(\theta) f, \]  

where \( M(\theta) \in \mathbb{R}^{n \times n} \) is the manipulator inertia matrix, \( C(\theta, \dot{\theta}) \dot{\theta} \in \mathbb{R}^n \) represents the centrifugal and Coriolis torques, \( g(\theta) \in \mathbb{R}^n \) represents the gravity
torques, $J^T(\theta)f \in \mathbb{R}^3$ being the vector of contact force exerted by the robot end-effector on the environment and $\tau \in \mathbb{R}^n$ is the vector of applied joint torques. It is well known that the left-hand side of (23) can be linearly parameterized by $Y(\theta, \dot{\theta}, \ddot{\theta})a$, where $a \in \mathbb{R}^m$ denotes the constant dynamic parameters and $Y \in \mathbb{R}^{n \times m}$ is the dynamic regressor matrix.

Here, the key idea is to introduce a cascade control strategy [6] to solve the hybrid velocity and force control problem for a robot manipulator with nonnegligible dynamics, analogous to the case that visual servoing problem was considered [21, 10]. To achieve this, one can assume that there exists a control law

$$\tau = F(\theta, \dot{\theta}, \ddot{\theta}, \dot{\theta}^d, \ddot{\theta}^d) - J^T(\theta)f,$$  

which guarantees the control goal defined by

$$\theta \rightarrow \theta^d(t), \quad e = \theta^d - \theta \rightarrow 0,$$  

where $\theta^d$ denotes the desired trajectory in the robot frame. Now, one supposes that it is possible to define the desired trajectory $\theta^d$ and its derivatives $\dot{\theta}^d, \ddot{\theta}^d$ in terms of a control signal $u$ such that one has (15) except for a vanishing term, that is,

$$\dot{x} = J(\theta)u + J(\theta)L(p)e,$$  

where $L(\cdot)$ denotes a linear operator with $p$ being the differential operator. Then, one can conclude that the kinematic control law (16) can be applied to (26).

Moreover, one can obtain some intuition if the parameters of the robot dynamic model (23) are assumed to be exactly known. A standard computed torque strategy could be used to solve the tracking problem, that is,

$$\tau = M(\theta)[\ddot{\theta}^d + K_d \dot{\theta}^d + K_p \dot{\theta}^d] + C(\theta)\dot{\theta} + g(\theta) - J^T(\theta)f,$$

driving a stable closed-loop system. Then, taking $\dot{\theta}_m = u$ one has that

$$\dot{x} = J(\theta)u + J(\theta)\dot{\theta},$$  

where $\dot{\theta}$ satisfies the closed-loop equation given by

$$\dot{\theta} = J(\theta)u + J(\theta)\dot{\theta}.$$  

This approach only differs from the kinematic control case by a vanishing term $\dot{\theta}(t)$. Thus, one can demonstrate that the kinematic control signal $u$, computed for the kinematic control case, can be applied to the case of dynamic robot control and the closed-loop stability can be proved.

Note that, in the case of parametric uncertainty on the robot system (23), adaptive or robust control strategies [16] can also be used for the dynamic robot control [21]. Furthermore, the proposed cooperative control scheme has passivity properties which make it possible to guarantee stability when cascaded to another adaptive control system with similar passivity properties as presented in [10, 13].

### 4 Experimental and Simulation Results

Both simulations and experiments are performed to verify the efficiency and stability of the proposed approach. Two robot manipulators hold an object while following a velocity trajectory generated by the master robots.

#### 4.1 Simulations

The control architecture is simulated with two master and two slave three joint robots implemented in Simulink. The masters are modeled as on [1], while the slaves consider only the Jacobian matrix according to (15). The communication channels are assumed lossless and with different time delays in the two directions, and the passivity and transparency layers are transmitted through the same channel. For simplicity the control loops on both sides run at the same 1 kHz rate using the same clock, so at each sample instant only one energy packet is received. On the master side a Cartesian velocity reference is tracked with a proportional-integral controller. The stiffness of the load transported by the slaves is 500 N/m, the control gain is $\Gamma = 5$ for both robots, and the desired force is set to $f^d = [0 \quad \pm 20]^T$.

On the slave side the passivity layer saturate force commands by the energy available in the tank with a mapping function

$$F_{max} = \sqrt{2H(k)k_{ss}}$$  

which acts like a linear spring with stiffness $k_{ss}$ and limits the force applied to the object as the energy in the tank builds up, as proposed in [5].

The complete set of parameters used in the energy tank implementation are listed in Table 1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma_m$</td>
<td>0.1</td>
</tr>
<tr>
<td>$H_d$</td>
<td>0.1</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>100</td>
</tr>
<tr>
<td>$\beta_s$</td>
<td>0.05</td>
</tr>
<tr>
<td>$k_{ss}$</td>
<td>1</td>
</tr>
</tbody>
</table>

The slave velocity trajectory tracking errors, when masters are describing a circle, are shown in Fig. 1. Both the robots could track the reference, therefore some distortions are observed due to the non-constant velocity reference and the coupled force control.

The influence of the energy tank control saturation could observed on both trajectories. The manipulators do not track the reference until have some amount of energy available.

Fig. 2 show the magnitude of the grasp forces during simulation. It is clear that the grasp can be maintained. Note that as stated in Theorem 1 grasp regulation sustain a bounded force, unless
on another simulation the master system moves with a constant velocity along $y$ direction. The grasp forces are showed on Fig. 3.

4.2 Experimental Setup

The slave robot used in the experiments is a dual arm (7 DOFs each) Motoman DIA10 controlled in low-level by a NX100 controller. The references are generated by external computer connected to the NX100 by a Motoman high speed controller (HSC). As is common on industrial robots the control loop is implemented on a kinematic level in joint space. As the robot does not have force sensors, a virtual object with a stiffness of 500 N/m and an ideal force sensor is implemented in the control loop.

In order to get access to the HSC controller from Matlab we use the communication protocol provided in Robot Raconteur (RR). Robot Raconteur is a library and architecture developed for robotic and automation applications with distributed communication over networks [20]. The designed server provides information about position and torque in joint space and the virtual forces, receives the desired joint positions, and implements a mid-level position control loop. This loop is designed in Simulink and then compiled to run in a separate thread on the server with a 2 ms sample time.

The master robot is a 6 DOFs Phantom Omni haptic device also accessed from Matlab through a RR node server. This provides the position in Cartesian space and receives force commands to be applied to the operator on a low-level loop running at 1 kHz. The velocity reference sent to the slaves are taken as the difference in position between two consecutive samples.

A second experiment was performed with the system autonomously tracking, without an operator, a helix trajectory

$$
\dot{x} = \begin{bmatrix} 0.04 \cos(\frac{4\pi}{400}k) & -0.03 \sin(\frac{4\pi}{400}k) & -0.005 \end{bmatrix}^T
$$

with $k$ representing the sample counter.

On both experiments no dead zone was implemented on the force control, prioritizing the grasp over the velocity tracking.

4.3 Experimental Results

Fig. 5 shows the virtual forces measured during the experimental verification. The control law maintains the interaction forces fluctuating near the desired value, showed as traced lines. Note that the desired force squeezes the object so even with a small deviation from the desired force the robot will still hold the load.

This force tracking error is expected since the force and velocity controls are coupled and arise as a result of the desired velocity being non-constant, as showed in the Fig. 6. However the error is bounded.

Fig. 7 shows the manipulators trajectory in the spatial motion experiment. As can be noted the final part of the trajectory is more deformed.
the stability, on delayed communication and possible package loss, of the resulting bilateral teleoperation system.

**Appendix**

Consider the closed loop complete system (20). We can divide the states into \( y_1 = [\xi \ z_1]^T \) and \( y_2 = [\zeta_c]^T \). The following Lyapunov function

\[
V(\xi, z_i, \zeta_c) = \sum_{i=1}^{N} P_i(z_i) - P_i(z_i^d) - (f_i^d)^T(z_i - z_i^d) + \frac{1}{2} \zeta_c^T M_c \zeta_c
\]

(29)

is considered as a candidate function to Theorem 4.14 from [7]. As \( V \) is positive definite and radially unbounded, the condition \( \alpha(y_1) \leq V(y_1, y_2) \leq \beta(y_1) \) is valid.

The derivative of \( V \) is given by

\[
\dot{V} = \sum_{i=1}^{N} [f_i^T \dot{z}_i - (f_i^d)^T \dot{z}_i^d - (f_i^d)^T z_i + (f_i^d)^T z_i^d] + \zeta_c^T M_c \dot{\zeta}_c
\]

\[= \sum_{i=1}^{N} (f_i - f_i^d)^T \dot{z}_i + \zeta_c^T \sum_{i=1}^{N} f_i - \zeta_c^T M_c \dot{\zeta}_c
\]

\[= \sum_{i=1}^{N} (f_i - f_i^d)^T \dot{\xi}_i - \zeta_c^T M_c \dot{\zeta}_c + \xi_c^T M_c \dot{\zeta}_c
\]

\[= \sum_{i=1}^{N} (f_i - f_i^d)^T K_i (f_i - f_i^d) - \xi_c^T M_c \dot{\zeta}_c\]

(30)

where the assumption (14) is considered. The second term on (30) has an undefined signal, so the boundedness cannot be directly concluded.

An upper bound can be obtained considering the bounded reference acceleration \( \dot{\zeta_c} \) and (19), as

\[
\dot{V} \leq \sum_{i=1}^{N} -k_1 \Vert \zeta_c \Vert^2 + k_2 \Vert \zeta_c \Vert \Vert \dot{\zeta}_c \Vert
\]

\[\leq \sum_{i=1}^{N} -k_1 \Vert \zeta_c \Vert^2 + k_3 k_1 \Vert \zeta_c \Vert \Vert \dot{\zeta}_c \Vert + k_4 \Vert \dot{\zeta}_c \Vert + k_5 \left( \Vert f_i \Vert + \Vert f_i^d \Vert \right)
\]

\[\leq \sum_{i=1}^{N} -k_1 \Vert f_i \Vert^2 + k_4 \Vert f_i \Vert + k_6 := W(y_1).
\]

(31)

were the constants \( k_1, k_4 \) and \( k_6 \) are

\[k_1 = \lambda_{max}(K_i), \quad k_4 = \frac{k_1 \Vert M_c \Vert \Vert \dot{\zeta}_c \Vert_{max}}{N}, \quad k_6 = \frac{k_4}{k_1} \Vert \zeta_c \Vert_{max} + k_1 \Vert f_i^d \Vert \left( \frac{k_4}{k_1} - \Vert f_i^d \Vert \right).
\]

The ultimate boundedness condition is that \( W(y_1) \) must be negative semi-definite out of a certain

**5 Conclusions**

This work presents a teleoperation scheme for multiple robots handling an object. The force closure is implemented on a kinematic level on the slave side to guarantee that the load is not dropped. In this sense, safety of the load is thus not affected by time delays and package losses. Furthermore, the master is allowed to apply additional forces on the load through the master manipulators and the resulting forces are feedback to the operator. Energy tanks are used to guarantee
ball on $y_1$-space. Observe that the roots from (31) are given by

$$\|f_i\| = \frac{k_4 \pm \sqrt{(k_4)^2 + 4k_1 \cdot k_6}}{2k_1}$$

with one positive and one negative solution. Then, the boundedness condition is $|f_i| > \frac{k_4}{k_1} + k$ where $k$ is a constant given by $k_1$ and $k_6$.

**Acknowledgment**

This work was supported in part by CNPq and FAPERJ, Brazil.

**References**


