

# Invariance Control in Robotic Applications: Trajectory Supervision and Haptic Rendering

Michael Scheint, Jan Wolff, Martin Buss  
Institute of Automatic Control Engineering  
Technische Universität München  
D-80290 München, Germany  
{m.scheint, jan.wolff, mb}@tum.de

**Abstract**—Invariance control is a novel approach from control theory for constrained control of nonlinear systems. This paper extends the invariance control method to multi-input systems subject to nonlinear constraints.

Experimental results of two robotic applications demonstrate the flexibility of the invariance control approach. The first scenario is trajectory supervision. The second one is haptic rendering of rigid surfaces. Experiments are conducted on a 3 DOF robot arm.

## I. INTRODUCTION

A standard problem in robot control is to restrict the position of the manipulator to its workspace, which size and shape are determined by the requirements of the application. This problem can be addressed by methods from robotics and constrained control.

In a path tracking scenario, collision avoidance can be achieved by a local strategy treating the problem as a control problem. Common approaches for collision avoidance are artificial potential field [1] methods, virtual force method [2] or escape velocities [3]. These methods avoid collisions with obstacles not only for the endeffector but for the complete robot arm.

In haptic rendering, the user perceives forces in response to interaction with objects of a virtual environment. Free-space motion and contact with rigid objects are the most challenging problems. The standard approach for rendering virtual walls is to switch to a spring-damper model on impact [4]. Impulse-based methods improve the realism of rigid objects by transmitting an impulse on impact [5]. While haptic rendering focuses on the perception of the virtual environment, rendering of rigid walls can be interpreted as a position constraint.

From the control theory perspective, the described robotic control problems are constrained control problems, where the state, the output and the input of a control system are subject to time-domain constraints. Established methods to address this problem are anti-windup/override control [6], the theory of constraint admissible/invariant [7], optimal control/model predictive control [8], the reference governor approach [9], backstepping using barrier functions [10] and invariance control [11], [12].

The basic idea of invariance control is to render a state space region positively invariant by switching the control input on its boundary. A subset of state-space is said to be

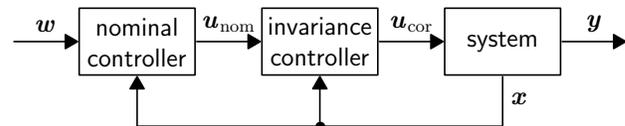


Fig. 1. Structure of Invariance Control

*positively invariant* with respect to a dynamical system if a trajectory with initial condition in the set remains therein for all future times [7]. Statements concerning invariance properties will subsequently always refer to positive invariance, as only the future system evolution is of interest for controller design.

Invariance control was originally developed for nonlinear systems with unstable internal dynamics, for example underactuated robots [13]. The switching control strategy allowed to control the output dynamics while keeping the internal dynamics bounded in an invariant set. Further research on invariance control for nonlinear control affine systems that are subject to state constraints lead to a different approach [11], which allowed to design an admissible invariant set from the constraints.

The basic structure of the control loop is depicted in Fig. 1. A nominal controller is assumed to achieve the main control objectives when the constraints are neglected. The invariance controller monitors the state of the system and modifies the control signal on the boundary of the invariant set. For single input systems, this can be interpreted as switching between two control modes: the nominal control mode and the corrective control mode. Consideration of multiple constraints requires to switch the scalar input between the nominal controller and different corrective controllers [12]. Continuous control mode transitions are investigated in [14], which increase the robustness with respect to parameter perturbations and disturbances. Other previous work includes control of humanoid robot knee with backlash [15] and balance control of a legged robot [16].

The main contribution of this paper is to extend invariance control to nonlinear affine systems with multiple inputs. In this case, switching in the control algorithm is replaced by an optimization problem. The requirement of invariance poses side conditions to the optimization problem. The objective function can be chosen freely. For two applications, results from experiments conducted on a 3 DOF robot arm are shown. In the first one, invariance control is used to

supervise a trajectory tracking controller. In the second one, invariance control is used for haptic rendering by displaying the constraints as stiff surfaces to the human operator.

The advantages of the proposed invariance control approach lie in its low computational effort for nonlinear systems and nonlinear constraints and its applicability to systems with an unknown future reference input. The last advantage makes the approach especially suitable for robotic scenarios with human interaction.

The remainder of the paper is organized as follows. In section 2, after giving the formal problem statement, the reader is introduced to the theory of invariance control for nonlinear multi-input systems under multiple constraints. The experimental setup is presented in section 3. In section 4, the trajectory supervision application is introduced and experimental results are presented. In section 5, the haptic rendering application is presented similarly. Finally, conclusions on the proposed control method are given in section 6.

## II. INVARIANCE CONTROL

### A. Notation

To ease notation, time derivatives of low order are denoted by  $\dot{y} := \frac{dy}{dt}$  and  $\ddot{y} := \frac{d^2y}{dt^2}$ , whereas time derivatives of higher order are denoted by  $y^{(i)} := \frac{d^i y}{dt^i}$ . The Lie-Derivative  $\mathcal{L}_f h(\mathbf{x})$  is the directional derivative of the scalar function  $h(\mathbf{x})$  in the direction of  $\mathbf{f}$ :

$$\mathcal{L}_f h(\mathbf{x}) = \frac{\partial h}{\partial \mathbf{x}} \mathbf{f}.$$

Higher-order Lie-Derivatives are defined recursively:

$$\mathcal{L}_f^0 h(\mathbf{x}) := h(\mathbf{x}) \quad \mathcal{L}_f^{i+1} h(\mathbf{x}) := \left( \frac{\partial}{\partial \mathbf{x}} \mathcal{L}_f^i h \right) \mathbf{f}$$

### B. Problem Statement

The considered class of dynamic systems are nonlinear control affine systems with order  $n$  and input dimension  $m$ :

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} = \mathbf{f}(\mathbf{x}) + \sum_{i=1}^m \mathbf{g}_i u_i. \quad (1)$$

$$\mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^m \quad \mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_m]$$

The functions  $\mathbf{f}$  and  $\mathbf{g}_i$  are smooth vector fields of appropriate dimension:  $\mathbf{f}, \mathbf{g}_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$ . A nominal controller  $\mathbf{u}_{\text{nom}} = \mathbf{k}_{\text{nom}}(\mathbf{x}, \mathbf{w})$  is assumed to exist which stabilizes (1) with respect to a reference trajectory  $\mathbf{w}(t)$ .

The state constraints are represented by a set of  $p$  nonlinear output functions  $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$ . A point  $\mathbf{x}$  in state-space is called admissible if all  $h_i$  are non-positive:

$$\forall_{1 \leq i \leq p} : \quad y_i = h_i(\mathbf{x}) \leq 0. \quad (2)$$

The maximal admissible set  $\mathcal{X}$ , which contains all admissible states, is the intersection of the zero-sublevel sets of the output functions  $h_i$ :

$$\mathcal{X} = \{ \mathbf{x} \mid h_i(\mathbf{x}) \leq 0, \forall i : 1 \leq i \leq p \} \quad (3)$$

Each output function  $h_i$  is required to have well defined relative degree  $r_i$  [17] in the maximal admissible set  $\mathcal{X}$ .

### C. Invariant Set Design

This section summarizes some of the results on invariance controller design from [11] and [12]. These results are extended to multi-input systems in the next section.

The maximal admissible set  $\mathcal{X}$  can only be kept invariant with finite control input for constraints with relative degree  $r = 1$ . For a single constraint, [11] describes a procedure to determine an admissible set  $\mathcal{G}$  that can be kept invariant by switching of the control on its boundary  $\partial\mathcal{G}$  for arbitrary relative degree. In the following, the index  $i$ , which specifies the respective constraints, is omitted for the sake of clarity.

The set  $\mathcal{G}$  is defined as the zero sublevel set of the invariance function  $\Phi(\mathbf{x})$ :

$$\mathcal{G} = \{ \mathbf{x} \mid \Phi(\mathbf{x}) \leq 0 \}. \quad (4)$$

Its boundary  $\partial\mathcal{G}$  is given the zero level set of  $\Phi(\mathbf{x})$ :

$$\partial\mathcal{G} = \{ \mathbf{x} \mid \Phi(\mathbf{x}) = 0 \}. \quad (5)$$

The invariance function is a function of the output  $y$  and its derivatives  $y^{(k)}$  with  $k < r$  and a design parameter  $\gamma$ . As  $r$  is the relative degree, the  $y^{(k)}$  can be calculated from the state  $\mathbf{x}$  alone. The value of  $\gamma$  is restricted to negative values for  $r \geq 2$  and  $\gamma = 0$  for  $r = 1$ . Larger absolute values for  $\gamma$  lead to larger admissible sets, but usually also to larger absolute values of the control input  $\mathbf{u}$ .

Explicit expressions for  $\Phi(\mathbf{x})$  can be calculated for low relative degrees, for example:

$$\begin{aligned} r = 1 : \quad & \Phi(\mathbf{x}) = y \\ r = 2 : \quad & \Phi(\mathbf{x}) = \begin{cases} y & \dot{y} \leq 0 \\ -\frac{1}{2\gamma}\dot{y}^2 + y & \dot{y} > 0. \end{cases} \end{aligned} \quad (6)$$

A main result of [11] is a sufficient condition for invariance of the set  $\mathcal{G}$ . The set  $\mathcal{G}$  is positively invariant for system (1), if at least one of the following two conditions is satisfied for each point on its boundary  $\partial\mathcal{G}$ :

$$\text{a) } \quad y^{(k)}(\mathbf{x}) < 0 \quad \text{for } 1 \leq k \leq r-1 \quad (7)$$

$$\text{b) } \quad y^{(r)}(\mathbf{x}, \mathbf{u}) \leq \gamma. \quad (8)$$

This condition exploits special properties of the invariance function  $\Phi(\mathbf{x})$  and allows to keep the set  $\mathcal{G}$  invariant by suitable choice of the control  $\mathbf{u}$  on the boundary such that condition b) is fulfilled wherever a) is not. Equation b) can be fulfilled by use of an input-output-linearizing control law.

Multiple constraints can be handled by designing a set  $\mathcal{G}_i$  for each of the constraints  $h_i$  separately and regarding a combined invariant set  $\mathcal{G}$  [12]. The set  $\mathcal{G}$  is defined as the zero sublevel set of the combined invariance function  $\Phi(\mathbf{x})$ :

$$\Phi(\mathbf{x}) = \max_i (\Phi_i(\mathbf{x})) \quad (9)$$

Thus,  $\mathcal{G}$  is the intersection of the sets  $\mathcal{G}_i$  and is admissible with respect to all constraints.

#### D. Determining Corrective Control

After having defined the invariant set  $\mathcal{G}$ , a corrective control input  $\mathbf{u}_{\text{cor}}$  is now derived which keeps the system invariant with respect to  $\mathcal{G}$ . This task consists of finding constraints for which condition (8) needs to hold and determining an input  $\mathbf{u}_{\text{cor}}$  which complies with the set of inequalities.

The first subtask can be dealt with using the concept of *active constraints*, i.e. outputs which have left or are about to leave the invariant set  $\mathcal{X}$  under nominal control. A constraint is active, if the state  $\mathbf{x}$  is in  $\mathcal{G}_i^+$ :

$$\mathcal{G}_i^+ = \left\{ \mathbf{x} \mid \Phi_i(\mathbf{x}) \geq 0 \wedge \exists_{1 \leq j \leq r_i-1} y_i^{(j)} > 0 \right\}. \quad (10)$$

The set of active constraints  $\mathcal{I}$  is then given as

$$\mathcal{I} = \left\{ i \mid \mathbf{x} \in \mathcal{G}_i^+ \wedge y_i^{(r)}(\mathbf{x}, \mathbf{u}_{\text{nom}}) > \gamma_i, 1 \leq i \leq p \right\} \quad (11)$$

containing those indices for which the invariance condition (8) needs to hold.

In order to obtain explicit conditions on  $\mathbf{u}$ , the derivatives of  $y_i$  are now analyzed for multiple inputs. From the relative degree assumption, it is known that the first  $r_i - 1$  derivatives  $y_i^{(k)}$  do not depend on  $\mathbf{u}$  [17]:

$$y_i^{(k)} = \mathcal{L}_f^k h_i(\mathbf{x}) + \underbrace{\left[ \mathcal{L}_{g_1} \mathcal{L}_f^{k-1} h_i(\mathbf{x}) \dots \mathcal{L}_{g_m} \mathcal{L}_f^{k-1} h_i(\mathbf{x}) \right]}_{=\mathbf{0}^T} \mathbf{u},$$

while the  $r_i$ -th derivative  $y_i^{(r)}$  does depend on  $\mathbf{u}$ :

$$y_i^{(r_i)} = \underbrace{\mathcal{L}_f^{r_i} h_i(\mathbf{x})}_{b_i(\mathbf{x})} + \underbrace{\left[ \mathcal{L}_{g_1} \mathcal{L}_f^{r_i-1} h_i(\mathbf{x}) \dots \mathcal{L}_{g_m} \mathcal{L}_f^{r_i-1} h_i(\mathbf{x}) \right]}_{\mathbf{a}_i^T(\mathbf{x})} \mathbf{u}.$$

Rewriting the set of equations (with not necessarily equal relative degrees  $r_i$ ) in matrix-vector notation we obtain

$$\mathbf{A}_{\mathcal{I}}(\mathbf{x}) \mathbf{u}_{\text{cor}} \preceq \mathbf{b}_{\mathcal{I}}(\mathbf{x}) \quad (12)$$

with  $\mathbf{A}_{\mathcal{I}}(\mathbf{x}) = [\mathbf{a}_i^T]$  and  $\mathbf{b}_{\mathcal{I}}(\mathbf{x}) = [\gamma_i - b_i(\mathbf{x})]$  with  $i \in \mathcal{I}$ . The operator  $\preceq$  denotes here the inequality  $\leq$  for all elements of the vectors. The existence of at least one solution  $\mathbf{u}_{\text{cor}}$  of (12), implying compliance with the constraints, can be ensured here by the following requirement on the outputs:

*Proposition 1:* The system (1) can be made invariant with respect to the set  $\mathcal{G}$ , if

$$\forall i, j \in \mathcal{I} \text{ and } \forall \mathbf{x} \in \mathcal{G} : \mathbf{a}_i^T \mathbf{a}_j \geq 0$$

holds with  $\mathbf{a}_i$  from (12).

*Proof:* Consider an input vector  $\mathbf{u}^*$  that is a linear combination of the vectors  $\mathbf{a}_i$ ,  $i \in \mathcal{I}$ .

$$\mathbf{u}^* = - \sum_{i \in \mathcal{I}(\mathbf{x})} \lambda_i \mathbf{a}_i \quad \lambda_i \in \mathbb{R}, \lambda_i > 0 \forall i$$

Hence, for all  $j \in \mathcal{I}$  the inequality  $\mathbf{a}_j^T \mathbf{u}^* < 0$  holds, as the sum contains at least the nonzero element  $\mathbf{a}_j^T \mathbf{a}_j$ . In matrix vector notation, this results in

$$\mathbf{A}_{\mathcal{I}}(\mathbf{x}) \mathbf{u}^* \prec \mathbf{0}$$

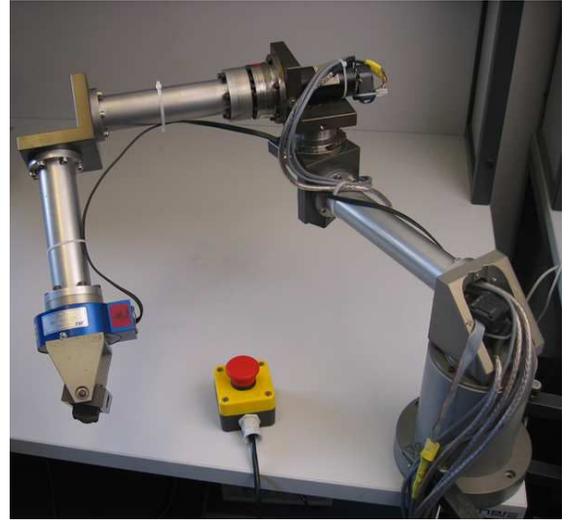


Fig. 2. The ViSHaRD3 device

Now it is possible to find a scalar  $\kappa$  such that the inequality

$$\mathbf{A}_{\mathcal{I}}(\mathbf{x}) \kappa \mathbf{u}^* - \mathbf{b}_{\mathcal{I}} \preceq \mathbf{0}, \quad \kappa > 0$$

holds and  $\mathbf{u}_{\text{cor}} = \kappa \mathbf{u}^*$  is a solution to (12). ■

As the problem of finding appropriate  $\mathbf{u}_{\text{cor}}$  can be underdetermined remaining degrees of freedom can be used to choose  $\mathbf{u}_{\text{cor}}$  such that it is equal to  $\mathbf{u}_{\text{nom}}$  in the unconstrained directions:

$$\begin{aligned} \min_{\mathbf{u}_{\text{cor}}} \|\mathbf{u}_{\text{cor}} - \mathbf{u}_{\text{nom}}\|_2^2 \\ \text{s.t. } \mathbf{A}_{\mathcal{I}}(\mathbf{x}) \mathbf{u}_{\text{cor}} - \mathbf{b}_{\mathcal{I}}(\mathbf{x}) \preceq \mathbf{0}. \end{aligned} \quad (13)$$

An advantage of this optimization approach is that  $\mathbf{u}_{\text{cor}}$  equals  $\mathbf{u}_{\text{nom}}$  if no constraints are active.

In the experimental setup described in the subsequent section, a robot arm with 3 DOF is subject to position constraints. Hence, a maximum of three constraints can be simultaneously active, i.e.  $\dim \mathbf{u} \geq \dim \mathbf{b}_{\mathcal{I}} \forall \mathcal{I}$ . Then, if  $\mathbf{A}_{\mathcal{I}}$  has full row rank, an analytical solution exists for (13). For the projection of  $\mathbf{u}_{\text{nom}}$  on the polyhedral set of side conditions in (13), the solution is found on the boundary of the polyhedron, where invariance is still guaranteed:

$$\mathbf{u}_{\text{cor}} = \mathbf{A}_{\mathcal{I}} \left( \mathbf{A}_{\mathcal{I}} \mathbf{A}_{\mathcal{I}}^T \right)^{-1} (\mathbf{b}_{\mathcal{I}} - \mathbf{A}_{\mathcal{I}} \mathbf{u}_{\text{nom}}) + \mathbf{u}_{\text{nom}}. \quad (14)$$

### III. EXPERIMENTAL SETUP

The robot used in our experiments is the ViSHaRD3<sup>1</sup>, a 3 DOF arm with fixed end-effector shown in fig. 2. The three revolute joints are each actuated by a Maxon RE40 DC motor coupled with a harmonic drive gear (gear ratio 1:100). Digital encoders HEDL-5540 (2000 counts per turn) mounted on the motor side measure the joint angles. A 6 axis JR3 force-torque sensor measures force interaction from an operator at the end-effector for haptic rendering.

All experimental models were implemented in Matlab/Simulink using the RTAI target for the Realtime Workshop. The dynamic model of the ViSHaRD3 robot arm was

<sup>1</sup>Virtual Scenario Haptic Rendering Device

obtained using the Autolev software package and integrated into the simulink models via s-functions. The control algorithms run on an AMD Athlon 64X2 5200+ computer equipped with a Sensoray 626 and a JR3 measurement card. The sampling rate was set to 2 kHz for all the experiments.

#### IV. TRAJECTORY SUPERVISION

In the first application in this paper, the goal is to keep the end-effector of the ViSHARD3 within an admissible region of the operational space while it follows a reference trajectory. In the following, first the system description of the manipulator under resolved acceleration control is derived. Then the nominal controller is introduced, which achieves the tracking objective. The chosen set of imposed position constraints is shown, for which invariance control is applied. Finally, experimental results are discussed.

##### A. System description

The end-effector is position controlled in the cartesian world coordinate system  $\mathbf{x}$  using resolved acceleration control [18]. The Lagrangian representation of the robot dynamics is:

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}). \quad (15)$$

Here,  $\mathbf{q} \in \mathbb{R}^3$  denotes the vector of joint angles,  $\boldsymbol{\tau} \in \mathbb{R}^3$  the actuation torques,  $\mathbf{M}$  denotes the joint inertia matrix and  $\mathbf{n}$  represents the nonlinear influences. The relationship between the joint and end-effector velocities and accelerations is given by the Jacobian matrix  $\mathbf{J}$ :

$$\dot{\mathbf{q}} = \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{x}} \quad (16)$$

$$\ddot{\mathbf{q}} = \frac{d}{dt}(\mathbf{J}^{-1}(\mathbf{q}))\dot{\mathbf{x}} + \mathbf{J}^{-1}(\mathbf{q})\ddot{\mathbf{x}}. \quad (17)$$

Resolved acceleration control computes a torque input from a commanded cartesian acceleration using (17) and (15). The commanded cartesian acceleration  $\ddot{\mathbf{x}}_c = \mathbf{u}$  is then the input to a linear system:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \ddot{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \mathbf{u}. \quad (18)$$

##### B. Nominal Controller

One way to design a trajectory tracking controller in cartesian coordinates uses a simple PD Controller:

$$\mathbf{u}_{\text{nom}} = \ddot{\mathbf{x}}_c = \mathbf{K}_P(\mathbf{x}_d - \mathbf{x}) + \mathbf{K}_D(\dot{\mathbf{x}}_d - \dot{\mathbf{x}}) + \ddot{\mathbf{x}}_d. \quad (19)$$

Here,  $\mathbf{x}_d$ ,  $\dot{\mathbf{x}}_d$ ,  $\ddot{\mathbf{x}}_d$  denote the reference motion in cartesian coordinates and  $\ddot{\mathbf{x}}_c$  denotes the commanded acceleration to the resolved acceleration scheme.  $\mathbf{K}_P$  and  $\mathbf{K}_D$  are diagonal matrices containing the respective controller gains with  $\mathbf{K}_P = k_p \cdot \mathbf{I}$ ,  $\mathbf{K}_D = 2\sqrt{k_p} \cdot \mathbf{I}$  and  $k_p = 200$ .

##### C. Constraints

The set of constraints chosen here form a 90° segment of a cylindrical wall, which shall resemble a typical translational workspace of an articulated manipulator with two parallel rotary axes. The corresponding output functions are given in the table below.

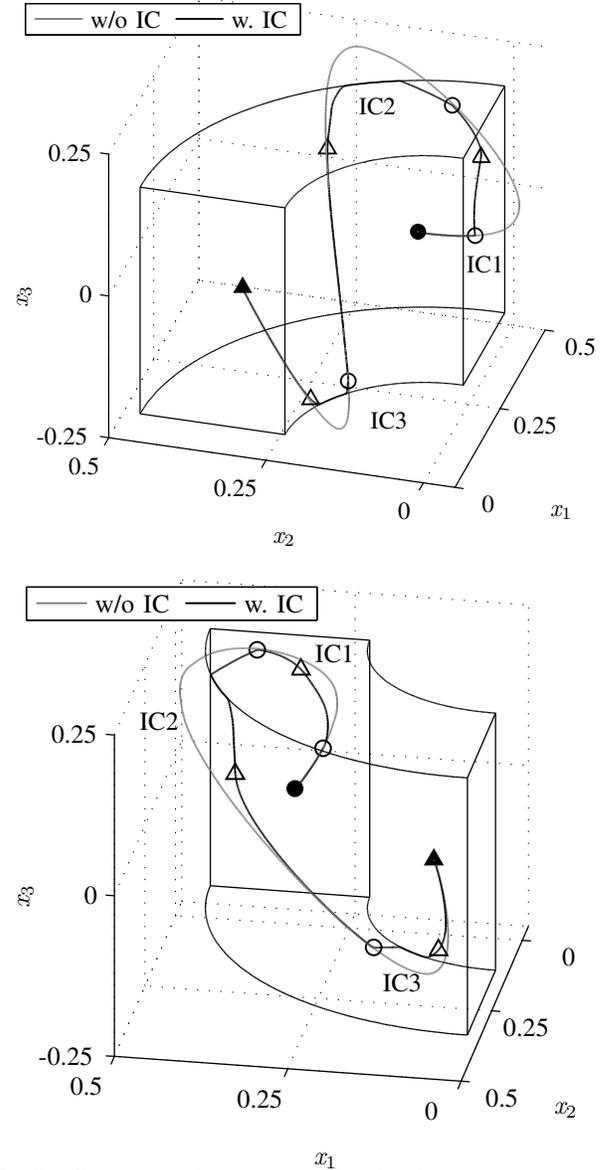


Fig. 3. Comparison of trajectories with and without invariance control.

left wall:	$y_1 = -x_1 + \hat{y}_1$	(20)
right wall:	$y_2 = -x_2 + \hat{y}_2$	
top wall:	$y_3 = x_3 + \hat{y}_3$	
bottom wall:	$y_4 = -x_3 + \hat{y}_4$	
inner radius:	$y_5 = -r^2 + \hat{y}_5$	
outer radius:	$y_6 = r^2 - \hat{y}_6$	

As the input to our system is a desired acceleration, the relative degree of the outputs is 2. Twice differentiating gives the corresponding  $\mathbf{a}_i$  and  $\mathbf{b}_i$  which can be used to solve (14) for  $\mathbf{u}_{\text{cor}}$ . The design parameters  $\gamma_i$  are chosen to have an absolute value of 90% of the torque saturation of  $8\frac{1}{2}$  Nm.

##### D. Stability

The sufficient condition for stability from [11] is not applicable, as the system trajectory cannot follow the reference trajectory outside the invariant set  $\mathcal{G}$ . Once the reference reenters the invariant set, nominal control asymptotically

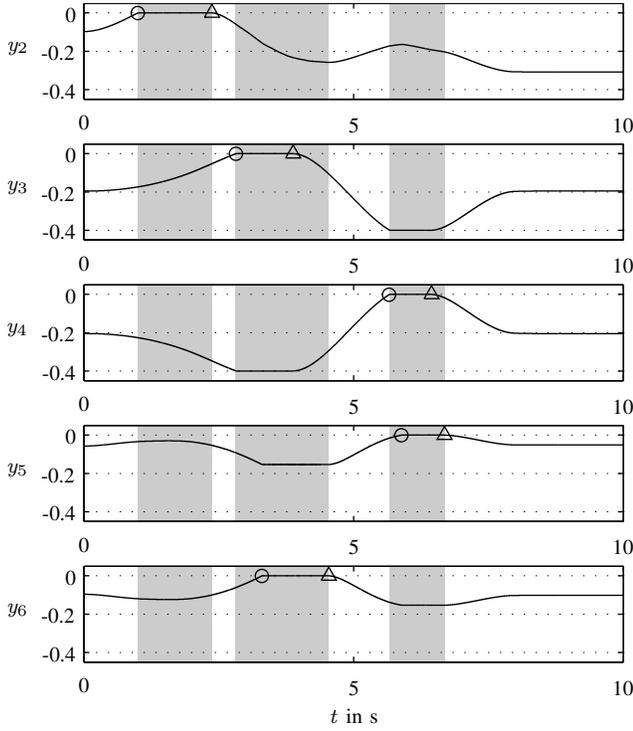


Fig. 4. Plot of output functions against time for trajectory supervision

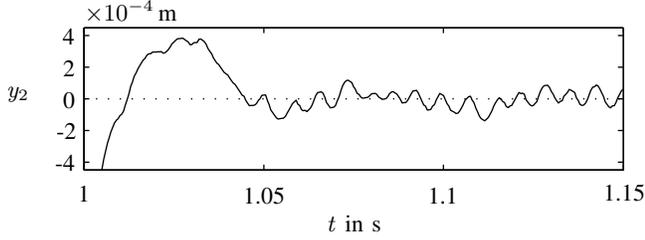


Fig. 5. Detail of output  $y_2$  at the start of corrective control.

stabilizes the system with respect to the reference trajectory.

### E. Experiment

The reference trajectory in the experiment is composed of cubic splines and partially leaves the admissible set. Therefore, constraint violations take place for the system under nominal control.

In Fig. 3 two 3D views of the trajectory of the end-effector are shown. The trajectory of the system under nominal control and invariance control are depicted in gray and black. The intersections of the constraint surfaces are illustrated by black lines as well. The starting and end points are marked by a filled circle and triangle, respectively. The time spans when invariance control is active are marked by hollow circles and triangles and labeled IC1, IC2 and IC3.

The 3D plots in Fig. 3 show that the system under invariance control does not leave the admissible set and tracks the trajectory inside the admissible set. On the boundary, the invariance controlled system follows the projection of the reference onto the constraint surface(s). An exception occurs, if the reference trajectory approaches again the admissible set from outside, as it can be seen at the end of phase IC2. Nominal control input from the PD controller is then

constraint admissible through the error in velocity prior to reference position being constraint admissible. Hence, a tracking error is induced which vanishes with time.

In Fig. 4 the plots of those output functions are shown which are once active. Circles and triangles mark the beginning and end, respectively, of the time span when the corresponding output is active. The time spans with invariance control being active are depicted by gray shaded areas being identical with those spans marked by IC1, IC2 and IC3 in Fig. 3. The system stays invariant as the outputs are kept within the zero sublevel set.

In Fig. 5, the output function  $y_2$  is shown when reaching the boundary of the admissible set. This output represents the worst case of all outputs. An overshoot of approx.  $4 \cdot 10^{-4}$  m and a subsequent chattering around zero can be observed. The overshoot and chattering result from model inaccuracies and digital implementation and would not occur for an exact model with infinite fast switching.

## V. HAPTIC RENDERING

The second application uses constraints to create a virtual environment for a haptic display. A human operator of the haptic device perceives rigid walls on the virtual surfaces specified by position constraints. In the admissible region, an admittance controller, which reads force input to generate a position reference, is used to render free space. In the following, first the concept of position based admittance control is introduced. Then, the constraints which generate the virtual environment in the experiment are stated. Finally, experimental results are discussed.

### A. Admittance Control

In general, an admittance is defined by a dynamic relationship between force and velocity. One possibility to define such a relation uses (diagonal) matrices  $\mathbf{M}_{VE}$  and  $\mathbf{K}_{VE}$  describing the translational mass and damping, respectively, of a mass-damper system. Then the relation between force and reference velocity is given by

$$\mathbf{f} = \mathbf{M}_{VE} \dot{\mathbf{x}}_d + \mathbf{K}_{VE} \mathbf{x}_d. \quad (21)$$

For rendering free space the inertia matrix was chosen as  $\mathbf{M}_{VE} = 3 \cdot \mathbf{I} \text{ kg}$  with a damping of  $\mathbf{K}_{VE} = 5 \cdot \mathbf{I} \frac{\text{kg}}{\text{s}}$ .

The inner position control loop of the haptic device is driven by the reference motion given in (21) which is generated through the operator force input. Not only reference position and velocity will be used in the position controller, but also the feedforward acceleration as it provides superior closed loop bandwidth [18].

$$\ddot{\mathbf{x}} = \mathbf{K}_P (\mathbf{x}_d - \mathbf{x}) + \mathbf{K}_D (\dot{\mathbf{x}}_d - \dot{\mathbf{x}}) + \ddot{\mathbf{x}}_d \quad (22)$$

The gain matrices  $\mathbf{K}_P = k_p \cdot \mathbf{I}$  and  $\mathbf{K}_D = 2\sqrt{k_p} \cdot \mathbf{I}$  of the PD position controllers were set to  $k_p = 500$ .

Furthermore, the resolved acceleration scheme used for the position control loop is adapted for haptic rendering. The commanded torque is adjusted to compensate the force  $\hat{\mathbf{f}}$  applied from the operator:

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}}_c + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) + \hat{\mathbf{J}}^T \hat{\mathbf{f}} - \mathbf{J}^T \mathbf{f}. \quad (23)$$

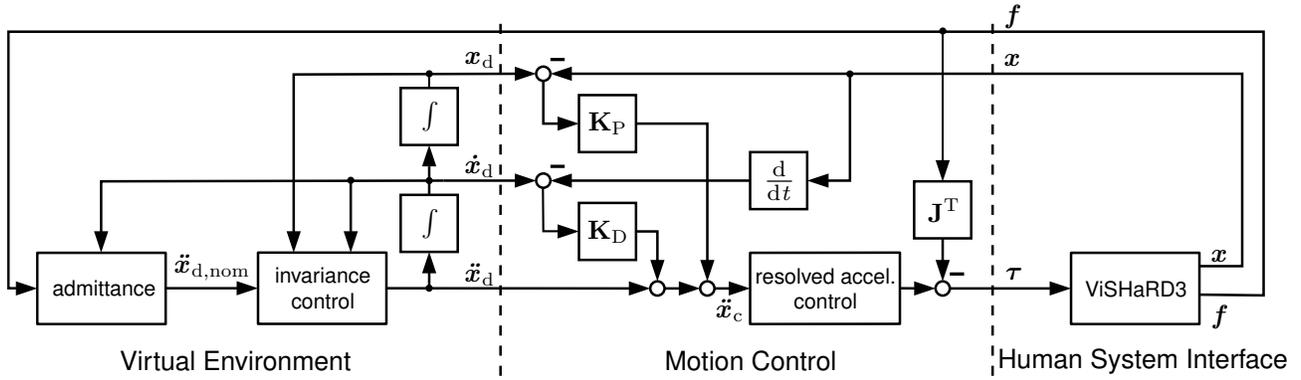


Fig. 6. Block diagram of the system under position based admittance control with an invariance controller.

Here,  $f$  denotes the measured force input from the user and the term  $\hat{\mathbf{J}}^T \mathbf{f}$  accounts for the induced joint torques from the force applied by the user.

If we assume a sufficiently good tracking of the position reference, then we can set  $\mathbf{x} \approx \mathbf{x}_d$ . Thus, the system we have to consider for invariance control is simplified to a system of double integrators similar to (18). In this case, the states are  $\mathbf{x}_d$ ,  $\dot{\mathbf{x}}_d$  and the input is the desired acceleration  $\ddot{\mathbf{x}}_d$ .

### B. Constraints

The virtual environment which shall be displayed to the human user in this application is a cuboid with a nonlinear bottom surface. This surface is chosen to be a two dimensional cosine wave as an example of a curved surface. Hence, six constraints are needed to specify the admissible region:

lower $x_{d,1}$ limit:	$y_1 = -x_{d,1} + \hat{y}_1$
upper $x_{d,1}$ limit:	$y_2 = x_{d,1} - \hat{y}_2$
lower $x_{d,2}$ limit:	$y_3 = -x_{d,2} + \hat{y}_3$
upper $x_{d,2}$ limit:	$y_4 = x_{d,2} + \hat{y}_4$
upper $x_{d,3}$ limit:	$y_5 = x_{d,2} - \hat{y}_5$
lower $x_{d,3}$ limit:	$y_6 = -x_{d,3} + \hat{y}_6$ $+ \mu \cos(\omega_1(x_{d,1} - x_{d,1, \min}))$ $+ \nu \cos(\omega_2(x_{d,2} - x_{d,2, \min}))$

Again, the second derivatives of the output functions give us the necessary terms which are needed to solve for (14). Here, the design parameters  $\gamma_i$  are independent of the torque saturation of 11 Nm and are set to  $\gamma_i = 40$ .

### C. Stability

For haptic rendering, the stability argument is based on passivity. It can be shown that invariance control is passive for position and velocity constraints, as the corrective control action always acts in the opposite direction of the velocity.

### D. Experiment

In the experiment conducted for haptic rendering, the human operator using the haptic device was moving the end-effector along the nonlinear surface. As the objective of the nominal controller is to render free space, only the constraints support the end-effector. The run is shown in Fig. 7, with the trajectory of the end-effector in black and the constraint  $y_6$  illustrated by contour lines in gray.

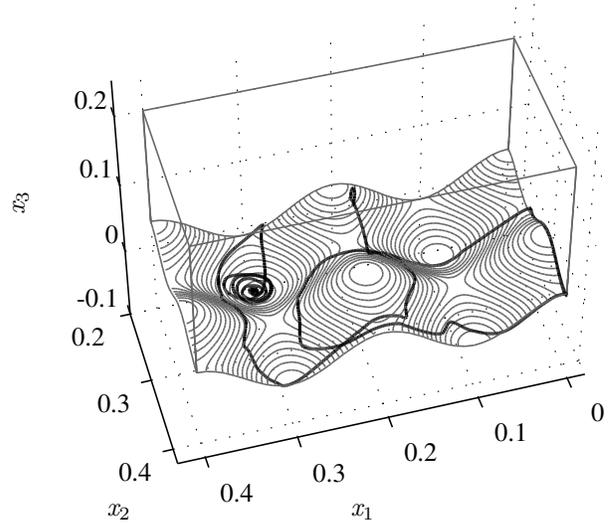


Fig. 7. Run of Position of end-effector during haptic rendering

During the run, between zero and three constraints were active, the latter case was given when the end-effector was in the right lower corner of the admissible space. This can be as well seen on the plots of the corresponding outputs, which are given in Fig. 8. It is important to note, that these plots show outputs as functions of the real coordinates  $\mathbf{x}$  and not of the reference coordinates  $\mathbf{x}_d$ , which are used by the invariance controller.

Fig. 9 highlights the differences between the output function  $y_6$  in the two coordinate systems. Whereas in the reference coordinates  $\mathbf{x}_d$  the output function shows only a chattering behavior of a magnitude mostly less than  $2.5 \cdot 10^{-5}$  m, in the real cartesian coordinates  $\mathbf{x}$  the output function trajectory lies in between  $-5 \cdot 10^{-3}$  m and 0 m. Concerning the latter value, it has to be noted that the steady state error of the position controlled robot is approx. 2 mm. Altogether, the system stays compliant with the constraints except for a few time instances corresponding to the peaks in the lower plot of Fig. 9.

In the last plot given in Fig. 10, the output function  $y_6$  is shown for the time span when the nonlinear surface is reached for the first time. The left plot shows an overshoot probably due to a force input by the operator exceeding the

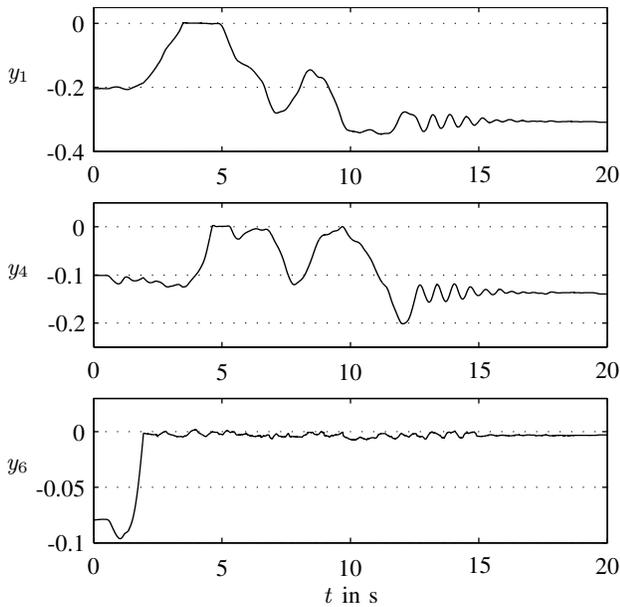


Fig. 8. Relevant output functions in haptic rendering run as seen in Fig. 7

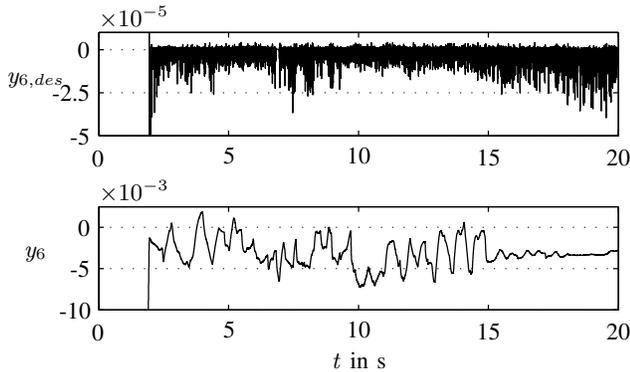


Fig. 9. Comparison of output function  $y_6$  in virtual and real coordinates “braking” force  $\gamma_6$ . However, the plot in the real coordinates shows no overshoot and the value lies within the order of magnitude of the steady state position error.

## VI. CONCLUSION

The proposed extension to the invariance control method allows constrained control of MIMO systems. The resulting control law guarantees compliance with the constraints. Its numerical simplicity makes it suitable for applications requiring high sampling rates. The presented examples of trajectory supervision and haptic rendering of rigid curved surfaces show the flexibility of the presented approach.

Future research is directed at extending robustness results from [14] to the MIMO case and a stability analysis of the presented applications.

## VII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the reviewers’ comments.

## REFERENCES

[1] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, vol. 2, Mar 1985, pp. 500–505.

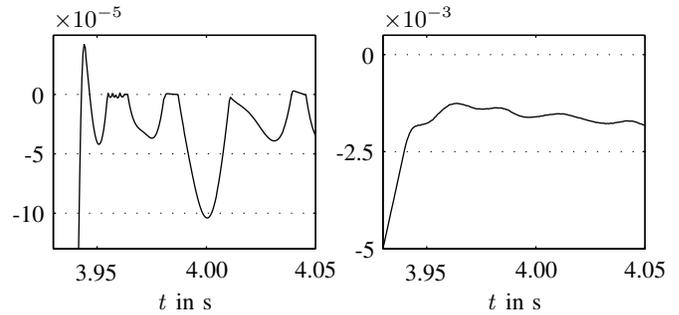


Fig. 10. Comparison of output function  $y_6$  in virtual (left) and real (right) coordinates upon impact.

- [2] H. Seraji and B. Bon, “Real-time collision avoidance for position-controlled manipulators,” *Robotics and Automation, IEEE Transactions on*, vol. 15, no. 4, pp. 670–677, 1999.
- [3] A. A. Maciejewski and C. A. Klein, “Obstacle Avoidance for Kinetically Redundant Manipulators in Dynamically Varying Environments,” *The International Journal of Robotics Research*, vol. 4, no. 3, pp. 109–117, 1985.
- [4] M. Ueberle, N. Mock, A. Peer, C. Michas, and M. Buss, “Design and Control Concepts of a Hyper Redundant Haptic Interface for Interaction with Virtual Environments,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, Workshop on Touch and Haptics*, Sendai, Japan, 2004.
- [5] E. Poorten and Y. Yokokohji, “Rendering a rigid virtual world through an impulsive haptic interface,” in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, Oct. 2006, pp. 1547–1552.
- [6] A. Glatfelter and W. Schauffelberger, *Control Systems with Input and Output Constraints*, ser. Advanced Textbooks in Control and Signal Processing. Springer-Verlag London, 2003.
- [7] F. Blanchini, “Set invariance in control – a survey,” *Automatica*, vol. 35, pp. 1747–1767, November 1999.
- [8] A. Bemporad, F. Borelli, and M. Morari, “Model predictive control based on linear programming – the explicit solution,” *IEEE Transactions on Automatic Control*, Vol. 47, Iss. 12, pp. 1974–1985, Dec. 2002.
- [9] A. Bemporad, “Reference governor for constrained nonlinear systems,” *IEEE Transactions on Automatic Control*, Vol. 43, Iss. 3, pp. 415–419, Mar. 1998.
- [10] K. Ngo, R. Mahony, and Z. Jiang, “Integrator backstepping using barrier functions for systems with multiple state constraints,” in *Proceedings of the IEEE Conference on Decision and Control*, Seville, Spain, Dec. 2005, pp. 8306–8312.
- [11] J. Wolff and M. Buss, “Invariance Control Design for Nonlinear Control Affine Systems under Hard State Constraints,” in *Proceedings of the NOLCOS 2004*, Stuttgart, Germany, Sept. 2004, pp. 711–716.
- [12] —, “Invariance Control Design for Constrained Nonlinear Systems,” in *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, July 2005.
- [13] J. Mareczek, M. Buss, and M. W. Spong, “Intelligent Switching Control of Nonlinear Non-minimum Phase Relative Degree Two Systems,” in *Proceedings of the European Control Conference*, Porto, Portugal, Sept. 2001, pp. 985–990.
- [14] C. Weber, J. Wolff, and M. Buss, “Continuous control mode transitions for invariance control of constrained nonlinear systems,” in *accepted for publication at the Conference on Decision and Control*, New Orleans, USA, Dec. 2007.
- [15] D. Wollherr, F. Zonfrilli, and Y. Nakamura, “Active-passive knee control for the humanoid UT-Theta,” in *Proceedings of the International Conference on Advanced Robotics*, Seattle, Washington, USA, 2005, pp. 692–697.
- [16] M. Sobotka, J. Wolff, and M. Buss, “Invariance controlled balance of legged robots,” in *European Control Conference ECC07. Proceedings of the*, Kos, Greece, July 2-5 2007, pp. 3179–3186.
- [17] A. Isidori, *Nonlinear Control Systems*, 3rd ed. London: Springer Verlag, 1995.
- [18] M.-W. Ueberle, “Design, control, and evaluation of a family of kinesthetic haptic interfaces,” Ph.D. dissertation, Technische Universität München, 2006.