Task-Space and Null-Space Control Design for Robotic-Assisted Minimally Invasive Surgery

Rui Cortesão*, Walid Zarrad**, Philippe Poignet**, Olivier Company** and Etienne Dombre**

*University of Coimbra, Institute of Systems and Robotics, 3030 Coimbra, Portugal
**LIRMM-UMR CNRS, University of Montpellier II, F-34392 Montpellier cedex 5, France
emails: cortesao@isr.uc.pt {zarrad, poignet, company, dombre}@lirmm.fr

***Abstract*— This paper discusses the control design of robotic-assisted minimally invasive surgery (MIS) with haptic feedback. The operational space control has a position-position telemanipulation architecture with the phantom in the loop, enabling telepresence in free-space and contact. The null space control guarantees that surgical kinematic constraints are fulfilled. Both task and posture control run active observers (AOBs) in Cartesian domain, taking into account force, velocity and position signals. Experiments with a D2M2 (direct drive medical manipulator) robot are presented[1].**

***Index Terms*— Medical robotics, haptics, active observers, task space, null space.**

I. Introduction

Robotic assisted surgery can greatly improve surgeons' skills. Dedicated instruments that enter in the human body through tiny holes can be tele-controlled through robotic systems, enhancing perception and execution of surgery tasks. Haptic feedback provides realism to contact interactions, distinguishing not only different objects, but also free-space to contact (and vice-versa) transitions. Scaling on-line tactile feedback can magnify or reduce haptic telepresence whenever needed, which is a key functionality for advanced surgery. Guaranteed stability in contact with stiff objects is an important milestone (e.g. touching another instrument or a rib for cardiothoracic surgery) which demands advanced control techniques, being a major obstacle for practical applications. Nowadays, robotized MIS does not include force feedback in the main control loop to avoid instability problems. The Da Vinci™ system from Intuitive Surgical is the state-of-the-art robotic system for MIS, in which surgeon hand movements are scaled, filtered and sent to the robotic end-effector. This setup has been used in many MIS, such as cardiac, urology and other abdomen surgeries, without haptic feedback. One main feature is a mechanically created fixed point that coincides with the penetration point. General purpose robotic manipulators can add this feature through proper control design, resorting from geometric models or null space functions. Although the problems of backlash and friction are highly reduced for direct-drive robots, nonlinear coupling among links are significant and the motor dynamics is complex. Therefore, good dynamic models and robust control techniques ought to be applied to achieve high performances.

[1]This paper has been published in part in [6].

The advantages over traditional (i.e., non-robotized) surgical practices include: 1) Augmented/scaled reality (e.g. motion and force augmentation or scaling for microsurgery); 2) Better comfort for the surgeon; 3) Real-time integration of intra-operative data (e.g. image-guided motion and force-controlled motion); 4) Accurate path following; 5) Enhanced mobility. Extra degrees of freedom inside the body can be controlled by the surgeon; 6) Compensation of physiological motions and surgical constraints; 7) Compensation for surgeon's hand tremor; 8) Less pain and trauma and shorter recovery time; 9) Tele-surgery; 10) Training, learning and mentoring using virtual models;

This paper proposes a systematic control design for robotized MIS, based on operational and null space techniques [9] linked to AOBs [4], [5]. It extends the work of [10] through haptic feedback, eliminating gradient-based functions from the control design.

Section II introduces the experimental setup and the D2M2 kinematic and dynamic models. Task space control is addressed in Section III, focusing on operational space techniques, feedback linearization and AOB design. Null space control is discussed in Section IV, considering specific constraints of MIS. Experimental results are presented in Section V. The conclusions are summarized in Section VI.

II. Experimental Setup

The master station has a Phantom 1.5 haptic device with six degrees of freedom (DOF) for position and force. The slave robot is the D2M2 robot designed for beating heart surgery experiments. It has five DOF with direct drive technology providing fast dynamics and low friction. An ATI force sensor is attached to the end-effector. The D2M2 is connected to a Pentium III at 500 [MHz] running under RTX/Windows 2000. The closed loop sampling time $h = 0.7$ [ms], and the system time delay

$$T_d = 5h \tag{1}$$

was obtained experimentally. Master and slave stations are connected via UDP communication under Windows XP. A picture of the experimental setup is represented in Fig. 1.

A. D2M2 Kinematic and Dynamic Models

The kinematic structure of the D2M2 is depicted in Fig. 2. The first joint is prismatic and the others are revolute with scara-like disposition. To optimize the overall weight and to

Fig. 1. Experimental setup. D2M2 robot tele-controlled by the Phantom for robotic surgery. The medical instrument has 40 [cm] height.

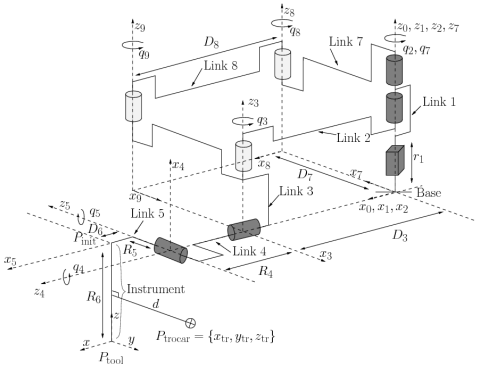

Fig. 2. D2M2 Kinematic model. Active, passive and cut joints.

boost the dynamic behavior, the motor for the third joint was placed on the prismatic axis, affecting the D2M2 models. Introducing the virtual parallelogram represented in Fig. 2 and also the active, passive and cut joint positions (q, q_p and q_c, respectively), the dynamic model which has contributions from active and passive joints is of form [8]:

$$M\ddot{q} + v(q,\dot{q}) + g(q) = \tau. \tag{2}$$

M is the mass matrix, $v(q,\dot{q})$ is the vector of Coriolis and centripetal forces, $g(q)$ is the gravity term and τ is the generalized torque acting on q. For the D2M2 robot, $g(q)$ is mechanically compensated, being not used in computational design. Further details, including numerical data, can be seen in [10]. In the experiments, the Symoro$^{\text{TM}}$ software has been used to compute the kinematic and dynamic models.

III. Task Space Control

A picture of the teleoperation scheme for the task space is represented in Fig. 3. The task is described in the base frame by 3D (three dimensional) Cartesian position/force vectors, associated to the D2M2 end-effector. The system plant $G(s)$ embeds operational space and feedback linearization techniques. The phantom position x_p scaled by β_p is compared with the end-effector position[2] X_t, generating a desired force $f_{i,t}$ through the virtual coupling K_v. To enhance the haptic

[2]The subscript "t" is used for task space variables, whenever appropriate.

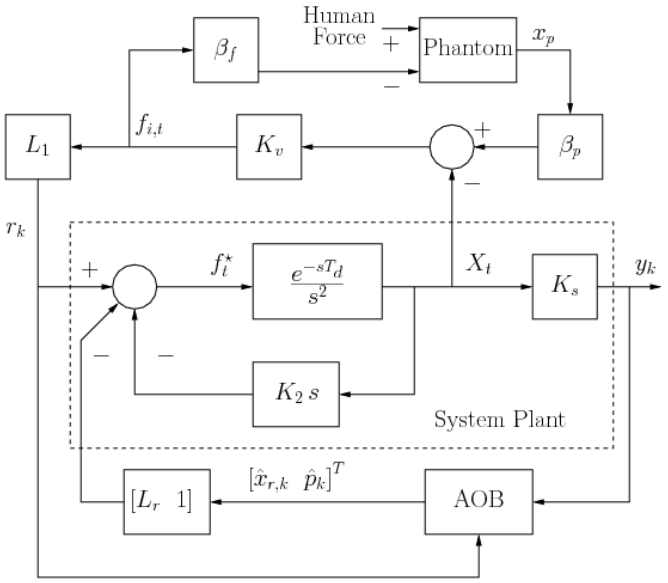

Fig. 3. Task space control scheme with AOBs for each Cartesian dimension. The master station, which includes the human and phantom, generates the 3D Cartesian force $f_{i,t}$ through the virtual coupling K_v. β_p scales the phantom position x_p, and β_f scales back $f_{i,t}$ to the master station. $G(s)$ has a damping term K_2, and is controlled by AOB estimates $\left[\hat{x}_{r,k} \;\; \hat{p}_k\right]^T$ through the state feedback gain $[L_r \;\; 1]$. K_s is the system stiffness and L_1 is the first element of L_r.

feeling quality of stiff objects, K_v may increase while in contact. A constant K_v establishes the trade-off between telepresence in contact, and robustness and comfortable performance in free space [5]. y_k is computed from force sensor measures projected into the base frame. It tracks the reference $f_{i,t}$ with desired dynamics through the AOB, even if the system stiffness K_s changes. $K_{s,n}$ is the value of K_s used in the control design, being estimated on-line from force data [5]. The human arm perceives $f_{i,t}$ scaled by β_f, which anticipates the real force. Telepresence is achieved if y_k follows well $f_{i,t}$. The main advantage of such position-position control scheme is that the human can feel position errors in free-space, due to motion, robot collision (not necessarily at the end-effector), or human-robot interaction. The main disadvantage is that impact events (high-frequency) at the end-effector are not accurately felt by the human (in this case, feeding back y_k would give better results). The AOB performs control actions only based on force signals (i.e., r_k and y_k).

A. Task Space Dynamics

Equation (2) for the operational space is

$$\Lambda_t \ddot{X}_t + V_t(q,\dot{q}) + g_t(q) = F_t, \tag{3}$$

with

$$\dot{X}_t = J_t \dot{q}, \tag{4}$$

$$\Lambda_t = (J_t^+)^T M J_t^+, \tag{5}$$

$$V_t(q,\dot{q}) = (J_t^+)^T v(q,\dot{q}) - \Lambda_t \dot{J}_t \dot{q}, \tag{6}$$

$$\tau_t = J_t^T F_t \tag{7}$$

and

$$g_t(q) = (J_t^+)^T g(q). \tag{8}$$

J_t, F_t and τ_t are respectively the Jacobian matrix, Cartesian force and task torque. J_t is a truncated non-squared matrix (3×5), therefore, J_t^+ represents its pseudo-inverse. In the experiments, the dynamically consistent pseudo-inverse has been used, i.e.

$$J_t^+ = M^{-1} J_t^T (J_t M^{-1} J_t^T)^{-1}, \tag{9}$$

which minimizes the robot kinetic energy [2], [8]. One advantage of truncated Jacobian solutions for robots that are not intrinsically redundant is the possibility to utilize the null space to optimize another objective function, very useful in robotic-assisted MIS. If external manipulation of the orientation is desired, a full Jacobian can be used, with zero values for the orientation vector (keeping the same task space controller)[3].

B. Feedback Linearization

Whenever the robot is in contact, an external force F_e appears at the end-effector. Hence, (3) can be written as

$$\Lambda_t \ddot{X}_t + V_t(q,\dot{q}) + g_t(q) = F_{c,t} + F_e, \tag{10}$$

where $F_{c,t}$ is the commanded force. For the desired Cartesian-decoupled system plant

$$\ddot{X}_t = f_t^\star, \tag{11}$$

$F_{c,t}$ should be[4]

$$F_{c,t} = -\hat{F}_e + \hat{V}_t(q,\dot{q}) + \hat{g}_t(q) + \hat{\Lambda}_t f_t^\star. \tag{12}$$

The estimation of F_e, $\hat{F}_e$, affects the control strategy [5]. The terms $\hat{V}_t(q,\dot{q})$, $\hat{g}_t(q)$ and $\hat{\Lambda}_t$ can be computed for a given robot. Introducing K_2, T_d and $K_{s,n}$, the desired system plant is[5]

$$G(s) = \frac{K_{s,n}\, e^{-sT_d}}{s(s + K_2\, e^{-sT_d})}. \tag{13}$$

Since T_d is small (see (1)),

$$G(s) \approx \frac{K_{s,n}\, e^{-sT_d}}{s(s+K_2)} \tag{14}$$

for a wide range of frequencies. Its equivalent temporal representation is

$$\ddot{y}(t) + K_2 \dot{y}(t) = K_{s,n} u(t - T_d), \tag{15}$$

where $y(t)$ is the plant output (Cartesian force at the robot's end-effector), and u is the plant input (force). Defining the state variables $x_1(t) = y(t)$ and $x_2(t) = \dot{y}(t)$, (15) can be written as

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -K_2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ K_{s,n} \end{bmatrix} u(t - T_\mathrm{d}). \tag{16}$$

[3]This solution is not appropriate for null space methods, though.

[4]The symbol ˆ means estimate.

[5]The analysis is done for each Cartesian dimension.

In compact form,

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t - T_d) \\ y(t) = x_1(t) \end{cases}. \tag{17}$$

Discretizing (14) with sampling time h [1], the equivalent discrete time system is

$$\begin{cases} x_{r,k} = \Phi_r x_{r,k-1} + \Gamma_r u_{k-1} \\ y_k = C_r x_{r,k} \end{cases}, \tag{18}$$

with

$$T_d = (d-1)h + \tau', \tag{19}$$

$$0 < \tau' \leq h, \tag{20}$$

$$x_{r,k} = \begin{bmatrix} x_k & u_{k-d} & \cdots & u_{k-2} & u_{k-1} \end{bmatrix}^T, \tag{21}$$

$$\Phi_r = \begin{bmatrix} \Phi_1 & \Gamma_1 & \Gamma_0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \tag{22}$$

$$\Gamma_r = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix}^T \tag{23}$$

and

$$C_r = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \end{bmatrix}. \tag{24}$$

Φ_1, Γ_0 and Γ_1 are given by

$$\Phi_1 = e^{Ah} = \phi(h), \tag{25}$$

$$\Gamma_0 = \int_0^{h-\tau'} \phi(\lambda)\, d\lambda B \tag{26}$$

and

$$\Gamma_1 = \phi(h - \tau') \int_0^{\tau'} \phi(\lambda)\, d\lambda B. \tag{27}$$

In our case, x_k has dimension seven. The first two states represent the force and force derivative, respectively. The other five states appear due to T_d. The continuous state transition and command matrices are

$$\phi(t) = \begin{bmatrix} 1 & \frac{1-e^{-K_2 t}}{K_2} \\ 0 & e^{-K_2 t} \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 \\ K_{s,n} \end{bmatrix}. \tag{28}$$

From (28), the computation of Φ_1, Γ_0 and Γ_1 is straightforward.

C. AOB Design

To accomplish model-reference adaptive control, the AOB reformulates the Kalman filter, based on [4], [5]:

1) A desired closed loop system (reference model) that enters in the state estimation.
2) An extra equation to estimate an equivalent disturbance referred to the system input, due to unmodeled terms including higher order dynamics, parameter mismatches and unknown disturbances. An active state p_k (extra state) describes the equivalent disturbance.

3) The stochastic design of the Kalman matrices for the AOB application. The first-order AOB algorithm is summarized in the sequel.

Controlling (18) through state feedback from an observer, and inserting p_k and $\hat{p}_k$ in the loop, the overall system can be represented by [5]

$$\begin{bmatrix} x_{r,k} \\ p_k \end{bmatrix} = \begin{bmatrix} \Phi_r & \Gamma_r \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{r,k-1} \\ p_{k-1} \end{bmatrix} + \begin{bmatrix} \Gamma_r \\ 0 \end{bmatrix} u_{k-1} + \xi_k \tag{29}$$

and

$$y_k = C_a \begin{bmatrix} x_{r,k-1} & p_{k-1} \end{bmatrix}^T + \eta_k, \tag{30}$$

where

$$u_{k-1} = r_{k-1} - \begin{bmatrix} L_r & 1 \end{bmatrix} \begin{bmatrix} x_{r,k-1} \\ \hat{p}_{k-1} \end{bmatrix}. \tag{31}$$

The stochastic inputs $\xi_k = \begin{bmatrix} \xi_{x_{r,k}} & {}^0w_k \end{bmatrix}^T$ and η_k represent respectively model and measure uncertainties. The state estimate of (29) is

$$\begin{bmatrix} \hat{x}_{r,k} \\ \hat{p}_k \end{bmatrix} = \begin{bmatrix} \Phi_r - \Gamma_r L_r & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{r,k-1} \\ \hat{p}_{k-1} \end{bmatrix} + \begin{bmatrix} \Gamma_r \\ 0 \end{bmatrix} r_{k-1} + K_k(y_k - \hat{y}_k), \tag{32}$$

with

$$\hat{y}_k = C_a \left(\begin{bmatrix} \Phi_r - \Gamma_r L_r & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{r,k-1} \\ \hat{p}_{k-1} \end{bmatrix} + \begin{bmatrix} \Gamma_r \\ 0 \end{bmatrix} r_{k-1} \right) \tag{33}$$

and

$$C_a = \begin{bmatrix} C_r & 0 \end{bmatrix}. \tag{34}$$

The Kalman gain K_k reflects the uncertainty associated to each state, which is a function of ξ_k and η_k [3], [7]. It is given by

$$K_k = P_{1k} C_a^T \left[C_a P_{1k} C_a^T + R_k \right]^{-1}, \tag{35}$$

with

$$P_{1k} = \Phi_n P_{k-1} \Phi_n^T + Q_k \tag{36}$$

and

$$P_k = P_{1k} - K_k C_a P_{1k}. \tag{37}$$

Φ_n is the augmented open loop matrix,

$$\Phi_n = \begin{bmatrix} \Phi_r & \Gamma_r \\ 0 & 1 \end{bmatrix}. \tag{38}$$

The system noise matrix Q_k is

$$Q_k = \begin{bmatrix} Q_{x_{r,k}} & 0 \\ 0 & Q_{p_k} \end{bmatrix}. \tag{39}$$

P_k and R_k are respectively the mean square error and measurement noise matrices.

D. Computed Torque for the Task

From (3), (7), (10), (12) and Fig. 3, the commanded torque for the task $\tau_{c,t}$ is

$$\tau_{c,t} = J_t^T \left\{ \hat{F}_e + \hat{V}_t(q,\dot{q}) + \hat{g}_t(q) + \hat{\Lambda}_t \left(r_k - \begin{bmatrix} L_r & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{r,k} & \hat{p}_k \end{bmatrix}^T - K_2 \dot{X}_t \right) \right\}. \tag{40}$$

In the experiments, $\dot{X}_t$ given by (4) was filtered by a discrete first-order low-pass filter. The estimate $\hat{F}_e$ is the first state of $\hat{x}_{r,k}$. The gravity $\hat{g}_t(q) = 0$ (see Section II-A).

IV. Null Space Control

For MIS, the task robot is teleoperated by a haptic device with force feedback, as explained in Section III. A secondary task is performed by a virtual null space robot, which attempts to have always the trocar position on the medical instrument. Virtual null space robots[6] can be defined by operational space techniques, short-cutting inverse kinematics problems, enabling the same control architecture of the task space. Although potential fields and associated gradient functions are very popular cost functions, this paper proposes another method to deal with MIS kinematic constraints, enabling posture control from 2D Cartesian positions only.

A. Null Space Robot

The difference between the null and task space robots represented in Fig. 4 is only at the tool length. The end-effector of the task space robot is in P_{tool}, with constant tool length. The virtual end-effector of the null space robot is in P_{null}, with time varying tool length, corresponding to the trocar projection onto the medical instrument. Both robots have the same dynamic model (the weight of the medical instrument is small). Only the kinematic models are slightly different. P_{null} is computed from

$$P_{\text{null}} = P_{\text{tool}} + (P_{\text{init}} - P_{\text{tool}})\rho \tag{41}$$

with

$$\rho = -\frac{(P_{\text{tool}} - P_{\text{trocar}}) \cdot (P_{\text{init}} - P_{\text{tool}})}{|P_{\text{init}} - P_{\text{tool}}|^2} \tag{42}$$

where $\cdot$ denotes the dot product and P_{init} is the tool starting point. To assess the null space robot behavior, the shortest distance between the medical instrument and trocar d_{tr} can be computed at each time step.

$$d_{\text{tr}} = \frac{|(P_{\text{init}} - P_{\text{tool}}) \wedge (P_{\text{tool}} - P_{\text{trocar}})|}{|P_{\text{init}} - P_{\text{tool}}|}, \tag{43}$$

where $\wedge$ denotes the cross product.

B. Control Design

The AOB control architecture for the null space is represented in Fig. 5. The robot is only position controlled in x_{null} and y_{null} (2D Cartesian coordinates), with a constant reference corresponding to the trocar position $\{x_{\text{tr}}, y_{\text{tr}}\}$. The virtual robot is synthesized by feedback linearization techniques in

[6]There is just one real robot that can be described by several virtual robots.

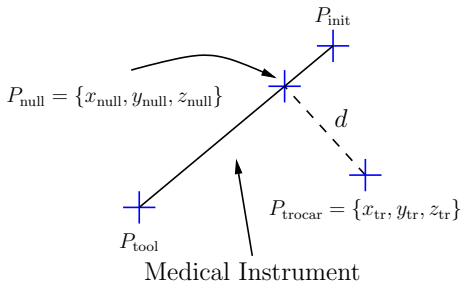

Fig. 4. End-effector of task and null space robots (P_{tool} and P_{null}, respectively). Trocar projection onto the medical instrument (P_{null}). P_{init} is the starting point of the medical instrument.

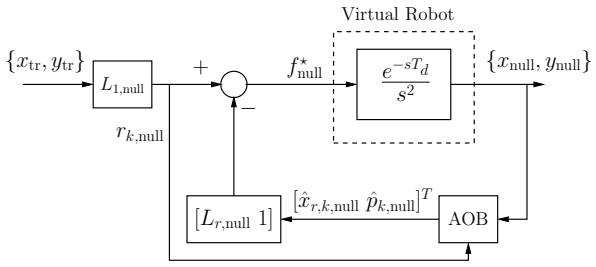

Fig. 5. Null space control scheme with AOBs for each Cartesian dimension. Position control architecture. $L_{1,\text{null}}$ is the first element of the state feedback gain $L_{\text{null}} = [L_{r,\text{null}}\ 1]$, The trocar position is the input and the trocar projection onto the medical instrument is the output.

the null Cartesian space[7]. The same procedure of (2)-(12) has been applied with minimal changes. The null Jacobian matrix J_{null} is non-squared (2×5), and takes into account the value of P_{null}. The system plant for each null Cartesian position is then

$$G_{\text{null}}(s) = \frac{e^{-sT_d}}{s^2}. \tag{44}$$

In the time domain[8]

$$\ddot{x}_{\text{null}} = f^{\star}_{\text{null}}(t - T_d). \tag{45}$$

Defining $x_{1,\text{null}}(t) = x_{\text{null}}$ and $x_{2,\text{null}}(t) = \dot{x}_{\text{null}}(t)$, (45) can be written as

$$\begin{bmatrix} \dot{x}_{1,\text{null}}(t) \\ \dot{x}_{2,\text{null}}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_{1,\text{null}}(t) \\ x_{2,\text{null}}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} f^{\star}_{\text{null}}(t - T_{\text{d}}), \tag{46}$$

The state space description of (46) has the same form of (16). Therefore, the subsequent analysis (including the AOB design) done for the task space control applies to null space as well. The computed torque for the null space robot $\tau_{c,\text{null}}$ is

$$\begin{aligned} \tau_{c,\text{null}} \;=\; & J^T_{\text{null}} \left\{ \hat{F}_{e,\text{null}} + \hat{V}_{t,\text{null}}(q,\dot{q}) + \hat{g}_{t,\text{null}}(q) - F_{i,\text{null}} \right. \\ & \left. + \hat{\Lambda}_{\text{null}} \left(r_{k,\text{null}} - [L_{r,\text{null}}\ 1]\,[\hat{x}_{r,k,\text{null}}\ \hat{p}_{k,\text{null}}]^T \right) \right\}, \end{aligned} \tag{47}$$

[7]The expressions of form "null *" mean "* for the null space robot".
[8]The analysis is done for x_{null}, which is the same for y_{null}.

TABLE I

AOB DESIGN PARAMETERS FOR EACH CARTESIAN DIMENSION OF BOTH TASK AND NULL SPACE ROBOTS. THE UNCERTAINTY ASSOCIATED TO EACH STATE IS $Q_{i,i}$ WITH $i = 1, \cdots, 8$. THE UNITS OF $Q_{i,i}$ AND R_k ARE STATE DEPENDENT.

	K_2	$K_{s,n}$	τ_d	$Q_{1,1}$	$Q_{i,i}$	$Q_{8,8}$	R_k
units	[Ns/m]	[N/m]	[s]				
x_t	5	300	0.05	10^{-9}	10^{-12}	10^{-3}	100
y_t	5	300	0.05	10^{-9}	10^{-12}	10^{-3}	100
z_t	5	300	0.05	10^{-9}	10^{-12}	10^{-3}	100
x_{null}			0.0035	0.7	10^{-12}	3500	100
y_{null}			0.0035	0.7	10^{-12}	3500	100

where $\hat{F}_{e,\text{null}}$, $\hat{V}_{t,\text{null}}(q,\dot{q})$ and $\hat{g}_{t,\text{null}}(q)$ are the 2D truncated versions of $\hat{F}_e$, $\hat{V}_t(q,\dot{q})$ and $\hat{g}_t(q)$, respectively. The induced Cartesian force on the null space robot due to the task $F_{i,\text{null}}$ is

$$F_{i,\text{null}} = \hat{\Lambda}_t f^{\star}_{t,\text{null}}, \tag{48}$$

corresponding to the 2D truncated value of $\hat{\Lambda}_t f^{\star}_t$. This force is compensated in (47). Projecting (47) in the task null space and adding (40), the total commanded torque τ_c is

$$\tau_c = \tau_{c,t} + (I - J_t^T (J_t^+)^T)\tau_{c,\text{null}}, \tag{49}$$

where I is the identity matrix [9].

V. EXPERIMENTS

This section reports tracking capabilities of the overall system subject to trocar constraints. It should be pointed out that at this design stage no real trocar has been inserted into the system. A trocar point was defined in space, along which the tele-controlled task was carried out. For the engineering design, system and AOB parameters have to be specified. In the experiments, $K_v = 1000$ [N/m], $\beta_p = 0.7$ and $\beta_f = 0.5$. L_r and $L_{r,\text{null}}$ have been computed by Ackermann's formula to achieve a critically damped response (for force and position, respectively) with desired time constant τ_d. The five additional poles (see Section III-B) were mapped at the origin (z-domain). Table I summarizes the AOB design. $K_{s,n}$ is kept low for free space, allowing fast motions. For contact tasks, the closed loop bandwidth is small, which is adequate for human manipulation. A higher null space bandwidth is required to avoid embarrassing values of d_{tr}. The stochastic estimation structure is more sensor based for the null space ($Q_{1,1}$ and $Q_{8,8}$ are higher), reacting faster to position errors. Both schemes follow model-reference adaptive control strategies, since the uncertainty associated with p_k ($Q_{8,8}$ value) is much higher than for the other states. The absolute values of Q_k and R_k are irrelevant for the AOB Kalman gain. Only relative relations are important [4].

A. Experimental Results

Figures 6 and 7 present tracking capabilites in free space with haptic feedback. Position errors are felt by the user, increasing motion perception. 3D tele-manipulation is well followed by the robot (Fig. 6), satisfying MIS kinematic

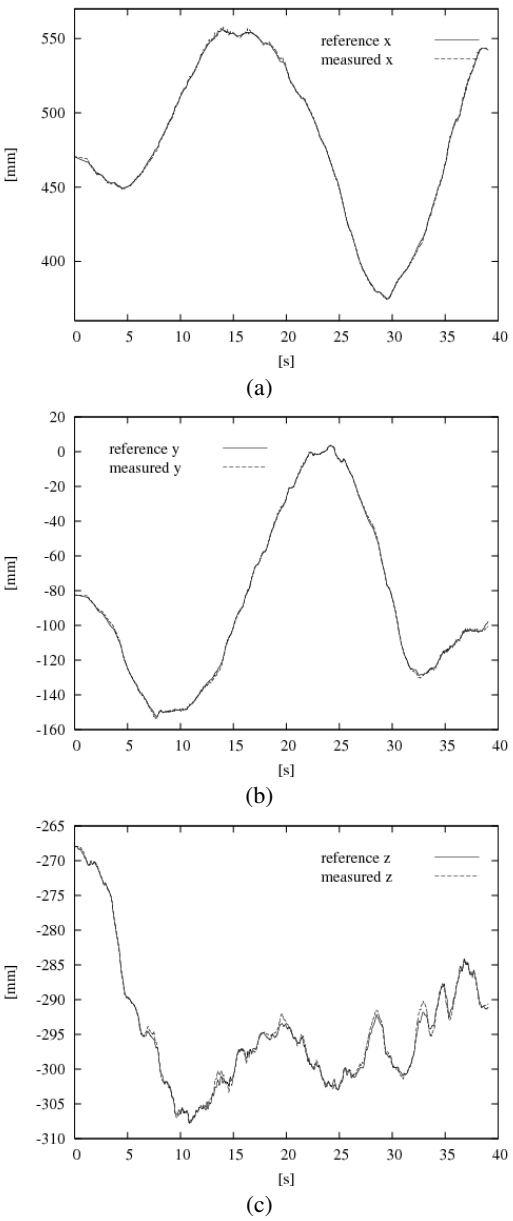

Fig. 6. Position tracking performance of the task space robot. 3D Cartesian motions tele-controlled by the phantom, satisfying trocar constraints.

constraints (Fig. 7). The root mean square errors (γ) in [mm] for the task are

$$\gamma(x_t) = 0.87, \quad \gamma(y_t) = 0.79 \quad \text{and} \quad \gamma(z_t) = 0.49, \tag{50}$$

and the mean value of d_{tr} is 2.62 [mm].

VI. Conclusions

This paper has presented a haptic control architecture for robotic-assisted MIS. The task space is tele-controlled by a human operator through a haptic device. A position-position teleoperation architecture has been used, where 3D Cartesian errors generate a desired force through virtual coupling. Tracking errors below 1 [mm] have been achieved, while

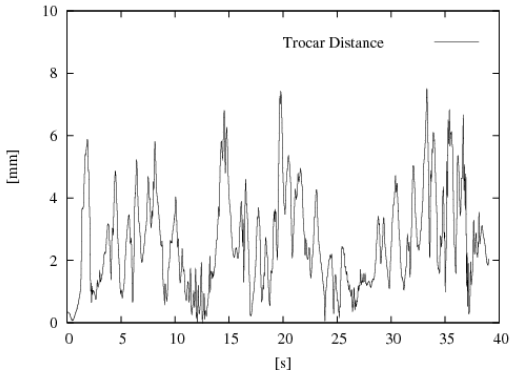

Fig. 7. Position tracking performance of the null space robot. Shortest distance between the medical instrument and trocar.

satisfying MIS constraints. A virtual null space robot has been introduced, enabling posture control to satisfy MIS kinematic constraints. 2D Cartesian position commands have been used for the posture, without resorting to gradient-based functions, inverse kinematics or explicit orientation commands. Induced Cartesian forces on the null space due to the task have been compensated. Null space position errors are below 3 [mm]. Both controllers apply AOBs, which run on top of operational space and feedback linearization techniques. Discrete state space methods, augmented states and stochastic state estimation belong to AOB design. For the engineering point of view, the control design is straightforward and the stochastic parameters provide enough flexibility to add dynamic functionalities without affecting control gains. Experimental results have shown good performance in free space motion subject to trocar constraints.

References

[1] K. J. Åström and B. Wittenmark. *Computer Controlled Systems: Theory and Design*. Prentice Hall, 1997.

[2] M. Benoit, M. Briot, H. Donnarel and. A. Liégeois, M. Meyer, and M. Renaud. Synthèse de la comande dynamique d'un téléopérateur redondant. *RAIRO*, pages 89–103, 1975.

[3] S. M. Bozic. *Digital and Kalman Filtering*. Edward Arnold, London, 1979.

[4] R. Cortesão. *Kalman Techniques for Intelligent Control Systems: Theory and Robotic Experiments*. PhD thesis, University of Coimbra, 2003.

[5] R. Cortesão, J. Park, and O. Khatib. Real-time adaptive control for haptic telemanipulation with kalman active observers. *IEEE Trans. on Robotics*, 22(5):987–999, October 2006.

[6] R. Cortesão, W. Zarrad, P. Poignet, O. Company, and E. Dombre. Haptic control design for robotic-assisted minimally invasive surgery. In *Proc. of the Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 454–459, China, 2006.

[7] A. Jazwinsky. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics In Science and Engineering*. Academic Press, 1970. (Edited by R. Bellman).

[8] W. Khalil and E. Dombre. *Modeling, Identification and Control of Robots*. Hermes Penton Ltd, 2002.

[9] O. Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *Int. J. on Robotics and Automation*, 3(1):43–53, February 1987.

[10] M. Michelin, P. Poignet, and E. Dombre. Dynamic task / posture decoupling for minimally invasive surgery motions. In *International Symposium on Experimental Robotics (ISER)*, Singapore, 2004.