2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Model Predictive Control Architectures with Force Feedback for Robotic-Assisted Beating Heart Surgery

Michel Dominici and Rui Cortesão

University of Coimbra, Institute of Systems and Robotics, 3030 Coimbra, Portugal
emails: michel.dominici@gmail.com cortesao@isr.uc.pt

***Abstract*— Minimally invasive surgery (MIS) offers considerable advantages for patients, lowering infection risks, and reducing trauma and convalescence times. Dedicated surgical robots significantly improve surgeon's skills especially for tasks requiring high precision such as lead placement and suturing. However, these robotic setups do not allow yet beating heart surgery with motion compensation functionalities. This paper tackles autonomous heart motion compensation with force feedback. We propose a cascade model predictive control (MPC) architecture with a Kalman Active Observer (AOB) in the loop, and compare it with the classical MPC approach. The cascade MPC-AOB control architecture has two loops. The inner one performs model-reference adaptive control, guaranteeing a desired force tracking dynamics. The outer one generates control actions to compensate physiological motions. Both MPC-based architectures are analyzed and experimentally evaluated. Two robots are used. A lightweight 4-DoF surgical robot generates desired surgical forces and a 3-DoF robot equipped with an ex vivo heart at the end-effector reproduces realistic heart motions.**

I. Introduction

In cardiac surgery, surgical procedures can be very problematic due to physiological motions. Performing complex, skillful and precise tasks on a moving organ, demands high-dexterous surgeons and enhanced concentration. Respiration is the most important source of disturbance, yielding large cyclic heart movements. Additionally, heartbeat motions involve high acceleration displacements. All of these disturbances cannot be manually compensated by the surgeon without phase and amplitude errors [1], [2], especially for accurate surgical procedures such as needle insertion, artery bypass graft, lead placement and suturing. Two classical techniques tackle this problem. An external machine can be used to substitute the heart while doing surgery. However, the use of a heart-lung system implies more risks and longer recovery times [3]. Another technique is based on a passive mechanical device called stabilizer. Placed around the workspace (e.g., coronary artery), it constraints motions by suction or pressure giving the idea of a quasi-static heart. Many improvements have been done since the first versions (in early 90's). Nevertheless, stabilizers still have several drawbacks [4] and [5]. For accessibility reasons, stabilizers can only be used for surgeries on the front of the heart. Additionally, intense pressure necessary to cancel motions may affect heart circulatory stability. Sucker-type stabilizers do not present this problem, but they introduce vacuum pressure that can rise to a level causing epicardial damage.

Robotic-assisted beating-heart surgery can efficiently circumvent surgical procedure difficulties through a proper control design. Compensating physiological motions autonomously through sensory data (e.g., vision and/or force) allows the surgeon to work on a virtual static organ. Architectures based on visual servoing have been first developed to compensate heart motions [6]. They proposed a hand support which tracks an interest point of the surgical site. A camera fixed to the support gives the distance information between the platform and the organ. Using this information, a control loop maintains the support at a constant distance. The surgeon's hand attached to the support is synchronized with the moving organ, while images from the camera system provide a stable view of the surgical site. More recently, Bachta and co-authors [7] proposed an active stabilizer to improve classical solutions. A common stabilizer has been piezo-electrically actuated to decrease residual heart motions. A high speed camera (frame rate: 333 [Hz]) tracks vertical displacements which have been used by an H_∞ control architecture. Solutions only based on visual servoing present drawbacks [8]. Minimally invasive surgery is performed in a cluttered environment where surgical instruments occlude artificial and natural landmarks. This situation entails tracking efficiency deterioration, disturbing the overall motion compensation. Moreover, contact tasks deform locally the surface of soft tissues (suturing, incision, ablation, etc.), affecting dramatically landmark calibration. Also, physiological motions induce disturbance forces during contact tasks that can hardly be compensated with vision information only.

Control architectures based on force information allow to manage interaction forces between tool and environment. In addition, force information may be used to give a haptic feedback to the surgeon. Kitagawa and co-authors [9] have quantified force feedback effects during precise surgery such as suturing. Experimental results have shown that force feedback improves robot-assisted performances during suture knot tying. Cagneau and co-authors [10] have proposed a force control architecture able to compensate periodic motions. An advanced iterative learning control have been implemented as an outer loop to reject periodic disturbances, reducing bad transients during the learning phase. No specific model is necessary for the robot and environment, although the perturbation period needs to be known in advance. This assumption is problematic for cardiac surgeries, due to random and chaotic nature of heart motions [11]. More recently,

978-1-4799-3684-7/14/$31.00 ©2014 IEEE

Kesner and Howe [12] have presented a catheter robotic system dedicated to beating heart surgery. A home made 1-DoF distal force sensor provides force feedback information. Additionally, a force-modulated position controller with friction and dead zone compensation was developed to apply a constant force on the mitral valve. Based on observations of previous cardiac motion cycles, a predictive autoregressive filter estimates the desired catheter acceleration, which is added to the control loop as a feedforward term. The results showed good capability to maintain a constant force on a fast moving target, although catheter-based solutions have a limited force range.

Based on the precedent work [13], we propose in this paper to improve the classical model predictive control architecture. We merge into two cascade loops the MPC approach presented in [14] with the AOB design described in [15], and compare it with the classical MPC architecture (cMPC). The new cascade control architecture is called MPC-AOB. Our predictive formulation is based on a pulse response model requiring an asymptotically stable plant. The inner loop based on a Kalman active observer guarantees an asymptotically stable system behavior. The external force control loop is based on a predictive formulation which has a mathematical model of the inner loop to predict system behavior. Both cMPC and MPC-AOB approaches are analyzed and assessed based on heart motion compensation capabilities. The paper is organized as follows. The task space formalism is given in Section II. Cascade MPC-AOB and cMPC architectures are discussed in Sections III and IV. Experimental results on physiological motion compensation are presented in Section V. Finally, Section VI concludes the paper.

II. Computed Torque Techniques in Task Space

Given a set of generalized joint coordinates q describing robot's posture, the well-known robot dynamics is represented by

$$M(q)\ddot{q} + v(\dot{q}, q) + g(q) = \tau, \tag{1}$$

where $M(q)$ is the mass matrix, $v(\dot{q}, q)$ is the vector of Coriolis and centripetal forces, $g(q)$ is the gravity term, and τ is the generalized torque acting on q. Using the operational space formulation, (1) can be written as

$$\Lambda_x(q)\ddot{X}_s + \Omega_x(q, \dot{q}) = F_c + F_e\,, \tag{2}$$

where X is the Cartesian position, $\Lambda_x(q)$ is the operational space mass matrix, and $\Omega_x(q, \dot{q})$ lumps Coriolis, centripetal, and gravity terms, all in Cartesian coordinates. F_c is the command force and F_e represents external forces acting on the robot end-effector. Knowing robot dynamic parameters and measuring F_e, F_c can be computed by

$$F_c = F_e + \Lambda_x(q) f^\star + \Omega_x(q, \dot{q})\,, \tag{3}$$

to obtained the decoupled plant[1]

$$\ddot{X}_s = f^\star. \tag{4}$$

[1]Modeling errors in (3) corrupt (4), motivating the use of the AOB architecture to compensate them.

Equation (4) represents the dynamics of a unitary mass. $f^\star$ is an acceleration, being an input parameter. Introducing a damping term K_2 and taking into account the system delay T_d (mainly due to signal processing), as well as a contact model[2] represented by $\hat{K}_s$ (an estimation of the heart stiffness K_s), the linear system plant G_{ol} for each Cartesian dimension is given by

$$G_{ol} = \frac{\hat{K}_s\, e^{-sT_d}}{s(s + K_2\, e^{-sT_d})}\,. \tag{5}$$

For small T_d,

$$G_{ol} \approx \frac{\hat{K}_s\, e^{-sT_d}}{s(s + K_2)}\,. \tag{6}$$

Its equivalent temporal representation is

$$\ddot{y}(t) + K_2\dot{y}(t) = \hat{K}_s v(t - T_d), \tag{7}$$

where $y(t)$ is the plant output (measured force at the robot's end-effector) and $v(t)$ is the delayed plant input. Defining the state variables $x_1(t) = y(t)$ and $x_2(t) = \dot{y}(t)$, (7) can be written as

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -K_2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \hat{K}_s \end{bmatrix} v(t - T_d). \tag{8}$$

Discretizing with sampling time T_s, the equivalent discrete time system is of form[3]

$$\begin{cases} x_{r,k} &= \Phi_r x_{r,k-1} + \Gamma_r v_{k-1} \\ y_k &= C_r x_{r,k} \end{cases}. \tag{9}$$

In our case, $x_{r,k}$ has dimension three. The first two states of the discretized system represent respectively the end-effector force and its derivative (only the force is measured). The other state is due to system delay $T_d = T_s$ and equal to v_{k-1}.

III. Cascade MPC-AOB Control Architecture

Due to a pole at the origin, the continuous plant (6) and its discrete equivalent (9) are not asymptotically stable. Even if classical MPC approaches can deal with this plant, well-defined stable plants are more robust to deal with external disturbances (such as heart motions). Therefore we have implemented an active observer to guarantee a stable plant.

A. Inner Loop

The AOB requires the description of the open loop system plant (9) as well as the desired closed loop model G_{cl}. Defining G_{cl} by

$$G_{cl} = \frac{1}{(1 + T_{cl}s)^2} e^{-sT_d}, \tag{10}$$

which corresponds to a critically damped system with time constant T_{cl}, a state-feedback L_r can be computed in straightforward way. Additionally, an extra state $\hat{p}_k$ is

[2]A more complex model taking into consideration the viscoelastic and anisotropic behavior of the myocardial tissue can be used (not addressed in this paper).

[3]See [16], [17] for further details on discrete matrices.

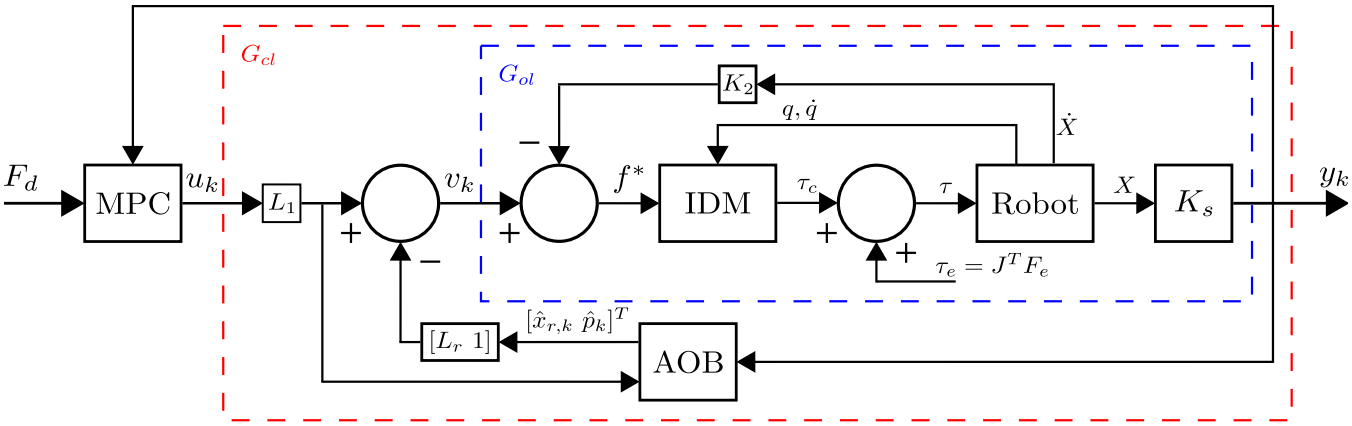

Fig. 1. Cascade MPC-AOB force control architecture for beating heart surgery. Computed torque techniques linked with the robot inverse dynamics model (IDM) generate a decoupled and linearized system. The open loop transfer function G_{ol} also takes into account a damping factor K_2 and the environment stiffness K_s. The desired closed loop transfer function G_{cl} is obtained by the AOB architecture using the state-feedback gain L_r and the extra state $\hat{p}_k$. L_1 is the first element of L_r. The MPC generates a processed reference force u_k for AOB control, based on the desired force F_d, the measured force y_k and G_{cl}. The external torque τ_e is mainly due to beating heart disturbances.

introduced to compensate system disturbances[4]. The AOB closed-loop estimation is given by[5]

$$\begin{bmatrix} \hat{x}_{r,k} \\ \hat{p}_k \end{bmatrix} = \hat{x}_a^- + K_k\,(y_k - \hat{y}_k) \tag{11}$$

and

$$\hat{y}_k = \begin{bmatrix} C & 1 \end{bmatrix} \hat{x}_a^-, \tag{12}$$

where the *a priori* augmented state estimation $\hat{x}_a^-$ is given by

$$\hat{x}_a^- = \begin{bmatrix} \Phi_r - \Gamma_r L_r & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{r,k-1} \\ \hat{p}_{k-1} \end{bmatrix} + \begin{bmatrix} \Gamma_r \\ 0 \end{bmatrix} u_{k-1}. \tag{13}$$

The Kalman gain K_k reflects uncertainties associated to the system state $\begin{bmatrix} \hat{x}_r(k) & \hat{p}(k) \end{bmatrix}^T$, depending on system ($Q_k$) and measurement ($R_k$) noise matrices. Q_k is of form

$$Q_k = \begin{bmatrix} Q_{x_{r,k}} & 0 \\ 0 & Q_{p_k} \end{bmatrix}. \tag{14}$$

The absolute values of R_k and Q_k are not important, since only the relative relation is relevant for the Kalman gain [15]. The AOB is inserted in the cascade architecture depicted in Fig. 1. The MPC generates a processed reference force u_k for AOB control, based on the prediction of the overall system dynamics, which includes the desired force F_d, the measured force y_k and G_{cl}.

B. Outer Loop

The MPC is a model based control architecture developed around a finite receding horizon strategy, and it requires a discrete state space model of G_{cl}. From (10), G_{cl} can be represented by

$$\begin{cases} x_k = Ax_{k-1} + Bu_{k-1} \\ y_k = Cx_k \end{cases}, \tag{15}$$

[4]Further details on AOB methodologies can be seen in [18], [17] and [15].

[5]A first-order AOB has been chosen.

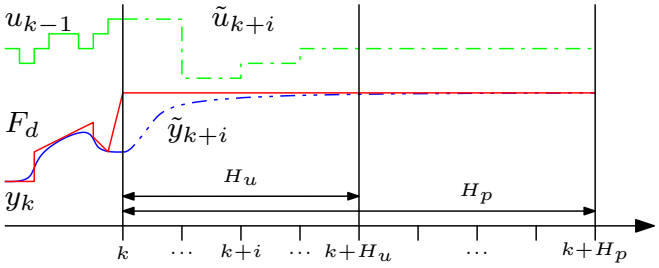

Fig. 2. Strategy of model predictive control. Based on G_{cl}, predicted outputs $\tilde{y}_{k+i}$ (blue dash-dot-dotted curve) are computed along H_p. The reference F_d is the desired force (red curve). Optimal control commands $\tilde{u}_{k+i}$ (green dash-dotted curve) are computed to minimize the cost function W.

where the small system delay T_d can be neglected for the MPC approach[6]. Therefore, x_k has dimension two representing the end-effector force and its derivative. y_k is the applied force. From (15), the x_k prediction i samples ahead, $\tilde{x}_{k+i}$, is based on u_{k-1}, y_k, $\tilde{u}_{k+i}$ and $\hat{x}_k$, where $\hat{x}_k$ is a state estimation of (15). Along the prediction horizon H_p, the future control sequence $\tilde{u}_{k+i}$ is computed by minimizing a cost function W to keep y_k as close as possible to the desired force F_d at time k, since future values of F_d are not known in advance for our system. Only the first element of the computed control sequence, $\tilde{u}_{k+1}$, is sent to the AOB control loop. As the prediction horizon is displaced towards the future, new output predictions $\tilde{x}_{k+i}$ and new control sequences $\tilde{u}_{k+i}$ are computed at each sampling time.

1) MPC Strategy: The methodology of the MPC is characterized by the following strategy:

- The finite time horizon H_p defines the slot where predicted outputs $\tilde{y}_{k+i}$ should follow F_d. H_p is bigger than the G_{cl} rise time, and its length greatly influences control tracking capabilities. Extending H_p improves

[6]Further details can be found in [19].

performance, but increases computational time.

- At each time k, based on (15), the future outputs $\tilde{y}_{k+i}$ are predicted along H_p, where $i \in [1, H_p]$. $\tilde{y}_{k+i}$ depends on $\hat{x}_k$, u_{k-1}, y_k and $\tilde{u}_{k+i}$.
- The command vector $\tilde{u}_{k+i}$ ($i \in [0, H_p-1]$) is computed to minimize the cost function W, which is a quadratic function of the predicted errors between $\tilde{y}_{k+i}$ and F_d. W also includes predicted control efforts. Two diagonal matrices λ and δ are associated to control efforts and tracking errors, respectively. The relation between λ and δ defines the aggressiveness of the controller in recovering from disturbances [20] and [21].
- A control horizon $H_u \leq H_p$ is introduced to reduce computation time. H_u defines the time slot along which the control command $\tilde{u}_{k+i}$ is active (for $H_u \leq i < H_p$, $\tilde{u}_{k+i}$ is kept constant). Although $H_u = 1$ has acceptable performance for stable plants, increasing H_u makes the control more active up to a limit where any further increase in H_u has little effect. For high-performance a larger value of H_u is desirable.
- Once $\tilde{u}_{k+i}$ has been computed, only the first element $\tilde{u}_{k+i|i=0}$ is applied to system. The whole cycle of output measurement y_k, output predictions $\tilde{y}_{k+i}$, and control sequence $\tilde{u}_{k+i}$ computation is repeated and updated at each sampling time.

The overall MPC strategy is depicted in Fig. 2. In our system, F_d is constant during the entire time horizon H_p. The MPC control signals $\tilde{u}_{k+i}$ are computed based on the system model and cost function W. A good control performance can be achieved with $H_u < H_p$, entailing good tracking capabilities between F_d and y_k.

2) MPC Formulation: From (15) and assuming $\tilde{u}_k$ is composed of two terms, the control increment $\Delta\tilde{u}_k$ and the previous command u_{k-1},

$$\tilde{u}_k = \Delta\tilde{u}_k + u_{k-1}, \tag{16}$$

we compute future states along the prediction horizon

$$\tilde{X_k} = \underbrace{\Psi\hat{x}_k + \Upsilon u_{k-1}}_{\text{past}} + \underbrace{\Theta\,\Delta\tilde{U_k}}_{\text{future}}, \tag{17}$$

with

$$\tilde{X_k} = \begin{bmatrix} \tilde{x}_{k+1} & \cdots & \tilde{x}_{k+i} & \cdots & \tilde{x}_{k+H_p} \end{bmatrix}^T, \tag{18}$$

$$\Delta\tilde{U_k} = \begin{bmatrix} \Delta\tilde{u}_k & \cdots & \Delta\tilde{u}_{k+H_u-1} \end{bmatrix}^T, \tag{19}$$

$$\Psi = \begin{bmatrix} A & \cdots & A^{H_u} & A^{H_u+1} & \cdots & A^{H_p} \end{bmatrix}^T, \tag{20}$$

$$\Upsilon = \begin{bmatrix} B \\ \vdots \\ \sum_{i=0}^{H_u-1} A^iB \\ \sum_{i=0}^{H_u} A^iB \\ \vdots \\ \sum_{i=0}^{H_p-1} A^iB \end{bmatrix} \tag{21}$$

and

$$\Theta = \begin{bmatrix} B & \cdots & 0 \\ AB+B & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{H_u-1} A^iB & \cdots & B \\ \sum_{i=0}^{H_u} A^iB & \cdots & AB+B \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{H_p-1} A^iB & \cdots & \sum_{i=0}^{H_p-H_u} A^iB \end{bmatrix}. \tag{22}$$

Equation (17) is composed of three terms. Ψ, Υ and Θ only depend on A and B matrices, and can be computed off-line. Along the prediction horizon, the first two terms represent the *free response* and the last term is the forced one. The increment sequence $\Delta\tilde{U_k}$ is computed by minimizing the cost function

$$W = (\tilde{Y_k} - \tilde{F_d})^T\delta(\tilde{Y_k} - \tilde{F_d}) + \Delta\tilde{U_k}^T\lambda\Delta\tilde{U_k}, \tag{23}$$

with

$$\tilde{Y_k} = \begin{bmatrix} C\tilde{x}_{k+1} & \cdots & C\tilde{x}_{k+i} & \cdots & C\tilde{x}_{k+H_p} \end{bmatrix}^T, \tag{24}$$

and

$$\tilde{F_d} = \begin{bmatrix} F_d & \cdots & F_d & \cdots & F_d \end{bmatrix}^T. \tag{25}$$

Defining the prediction error $\tilde{E_k}$ as the difference between $\tilde{F_d}$ and the *free response* of the system,

$$\tilde{E_k} = \tilde{F_d} - \mathrm{diag}(C)\left[\Psi\hat{x}_k + \Upsilon u_{k-1}\right], \tag{26}$$

the cost function (23) can be written as

$$W = (\Theta\Delta\tilde{U_k} - \tilde{E_k})^T\delta(\Theta\Delta\tilde{U_k} - \tilde{E_k}) + \Delta\tilde{U_k}^T\lambda\Delta\tilde{U_k}. \tag{27}$$

Developing (27), computing the gradient and set it to zero, we obtain the optimal and unique solution $\Delta\tilde{U_k}$ equal to

$$\Delta\tilde{U_k} = (\Theta^T\delta\Theta + \lambda)^{-1}\Theta^T\delta\tilde{E_k}. \tag{28}$$

Making $\delta \geq 0$ and imposing $\lambda > 0$ assures the Hessian of (27) is a positive-definite matrix, which is enough to guarantee that (28) is a the minimum. According to the MPC strategy previously described, the first element $\Delta\tilde{u}_{k+i|i=0}$ of the optimal increment sequence (28) is added to the previous command u_{k-1} and sent to the AOB control loop as u_k. All the computation is repeated at each sampling time.

IV. Classical MPC Control Architecture

The MPC-AOB controller is an improvement of previous work ([14]), where we presented a cMPC architecture dedicated to heart motion compensation. Both are based on the model predictive control algorithm presented above. This section highlights differences between cMPC and MPC-AOB approaches. cMPC is based on the discretized plant model (9) of G_{ol}, neglecting time delay[7]. Modeling errors are not compensated, corrupting the G_{ol} model which affects control

[7]The system state has dimension two representing the end-effector force and its derivative, and y_k is the applied force.

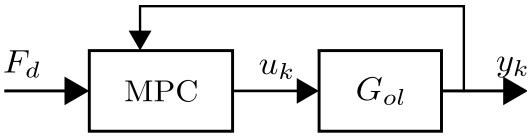

Fig. 3. cMPC force control architecture for beating heart surgery. G_{ol} represents the model of the unstable and linearized system plant. Based on a discrete G_{ol} model, desired forces F_d and applied forces y_k, the cMPC controller computes u_k to perform surgical tasks with autonomous heart motion compensation.

performance. From (9), future states are computed and equal to

$$\tilde{X_k} = \underbrace{\Psi \hat{x}_k + \Upsilon u_{k-1}}_{\text{past}} + \underbrace{\Theta \Delta \tilde{U}_k}_{\text{future}}, \tag{29}$$

with

$$\Delta \tilde{U}_k = \left[\begin{array}{ccc} \Delta \tilde{u}_k & \cdots & \Delta \tilde{u}_{k+H_p-1} \end{array} \right]^T, \tag{30}$$

$$\Psi = \left[\begin{array}{ccc} \Phi_r & \cdots & {\Phi_r}^{H_p} \end{array} \right]^T, \tag{31}$$

$$\Upsilon = \left[\begin{array}{ccc} \Gamma_r & \cdots & \sum_{i=0}^{H_p-1} {\Phi_r}^i \Gamma_r \end{array} \right]^T \tag{32}$$

and

$$\Theta = \left[\begin{array}{ccc} \Gamma_r & \cdots & 0 \\ \Phi_r \Gamma_r + \Gamma_r & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{H_p-1} {\Phi_r}^i \Gamma_r & \cdots & \Gamma_r \end{array} \right]. \tag{33}$$

Along the prediction horizon, the first two terms represent the *free response* and the last term is the forced one. Since the *free response* is based on the unstable system plant G_{ol} and the previous control command $u_{k-1} \neq 0$, the increment sequence $\Delta \tilde{U}_k$ has to be defined all over H_p, to maintain the system *free response* under "control". Therefore H_u is set equal to H_p. A large control horizon implies high dynamic control actions and increases computation time. As a consequence, H_p has to be decreased which affects tracking performance. The cMPC control architecture is depicted in Fig. 3. Based on G_{ol}, desired forces F_d and applied forces y_k, the cMPC generates a processed reference force u_k for the unstable and linearized system plant.

V. Experimental results

This section presents experimental results based on heart motion compensation capabilities. The robotic platform is presented in Fig. 4. It is composed by a lightweight 4-DoF WAM robot used as a tool holder, a 6-DoF JR3 force sensor[8] and a Heartbox. The Heartbox is a 3-DoF robot able to reproduce 3D motions with a precision of 0.1 [mm], where an *ex vivo* heart is attached and used as targets. The cMPC controller and the cascade MPC-AOB control described in Sections II, III and IV are implemented on a 2.1 GHz Intel Core 2 processor running Xenomai-Linux. The communication to the WAM robot is performed by CAN bus. Integrity of the WAM robot is checked through protection

[8]A Kalman filter is used to filter the force signal.

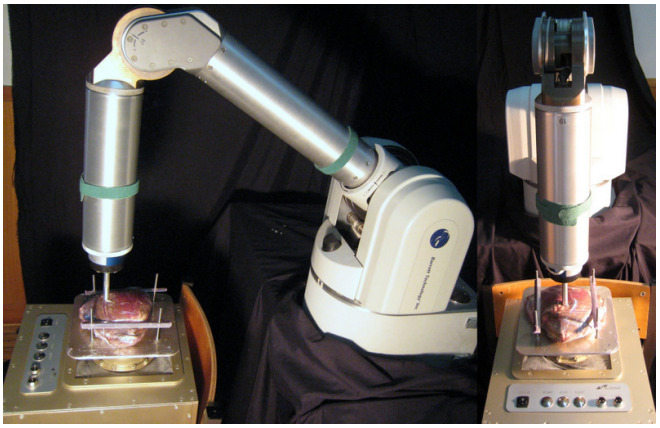

Fig. 4. Two views of the experimental setup. A lightweight 4-DoF WAM robot from Barrett Technology equipped with a 6-DoF JR3 force sensor interacts with a 3-DoF robot (called Heartbox) that generates 3D beating heart motions. An *ex vivo* heart is attached to the Heartbox and used as target for the WAM robot.

functions, such as maximum joint velocity, maximum joint torque, workspace limitation and maximum Cartesian forces. The control sampling time is set to 1 [ms].

A. Cascade MPC-AOB Control Design

1) Inner loop: Critically damped behaviors are appropriate for force-based tasks, since they represent the fastest response without overshoot. For a desired contact model $\hat{K}_s = 900$ [N/m], a damping $K_2 = 10$, and a desired G_{cl} given by (10) with $T_{cl} = 3$ [ms], the following state feedback gain

$$L_r = \left[\begin{array}{ccc} 161.5 & 1.189 & 0.557 \end{array} \right] \tag{34}$$

is obtained. The stochastic parameters reflect the model reference adaptive control strategy, where the uncertainties are lumped in p_k. R_k is set to 1 and Q_k is given by (see (14))

$$Q_{x_{r,k}} = \left[\begin{array}{ccc} 10^{-12} & 0 & 0 \\ 0 & 10^{-12} & 0 \\ 0 & 0 & 10^{-12} \end{array} \right] \tag{35}$$

and

$$Q_{p_k} = 0.5 \ . \tag{36}$$

This stochastic design entails the AOB Kalman gain

$$K_{k,f} = \left[\begin{array}{cccc} 0.1236 & 8.145 & 0.662 & 0.662 \end{array} \right]^T. \tag{37}$$

2) Outer loop: The length of the prediction horizon H_p greatly influences control tracking capabilities. Extending H_p, a more accurate system is achieved but the computational time increases. Since our control sampling time is $T_s = 1$ [ms], a good trade-off is achieved with $H_p = 30$ and $H_u = 5$. The optimal command $\Delta \tilde{u}_{k+i|i=0}$ is computed by minimizing the cost function W, with

$$\lambda = 0.1 I_u \tag{38}$$

and

$$\delta = 0.9 I_p, \tag{39}$$

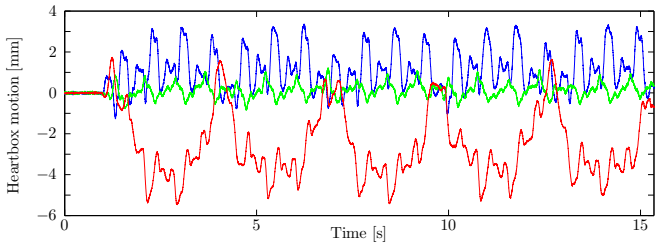

Fig. 5. Disturbances seen by the WAM robot and introduced by the Heartbox, X, Y and Z signals are represented by blue, green and red curves, respectively. The Heartbox is turned on after 1 [s].

where I_u and I_p are identity matrices of size H_u and H_p, respectively.

B. cMPC Control Design

Due to the unstable model G_{ol} to predict system outputs, the length H_u and H_p are equal. Extending the prediction horizon greatly increases control tracking capabilities leading to high dynamic control actions and a larger computational time. A good trade-off is achieved with $H_p = 12$. The matrix λ is increased to limit the control dynamics,

$$\lambda = 0.15 I_u, \tag{40}$$

which leads to

$$\delta = 0.85 I_p. \tag{41}$$

C. 3D Physiological Motion Compensation

Both control architectures are designed to compensate respiration and heartbeat disturbances autonomously. To evaluate compensation capabilities, respiration and heartbeat signals along three axes are generated by the Heartbox. They are based on physiological motion data recorded during *in vivo* experiments on a pig's heart [22]. The WAM robot applies forces on the moving heart attached to the Heartbox, and the goal is to track desired forces. Due to the soft nature of the heart, Heartbox motion seen by the WAM robot is significantly different. Fig. 5 represents the heart motion seen by the WAM robot along three axis. This motion is based on WAM robot Cartesian positions and residual force measurements recorded during two consecutive experiments: Heartbox turned on and off. Making spectral analysis, we can clearly identify two main sources of disturbance: breathing and heartbeat motions (see Fig 6). The first two peaks represent respiration (0.34 [Hz] and 0.72 [Hz]), corresponding to 20 breathing cycles per minute. The last five peaks are due to heartbeats (1.25 [Hz], 2.53 [Hz], 3.78 [Hz], 5.08 [Hz] and 6.32 [Hz]), which correspond to 75 heartbeats per minute.

Figs. 7 and 8 present 3D motion compensation results for cMPC and cascade MPC-AOB approaches, respectively. System force responses along three directions are represented separately. Colored curves represent applied forces and black curves are constant desired forces equal to 0 [N] for X and Y axes and -5 [N] for Z axis. The Heartbox is turn on after 1 [s]. The cascade MPC-AOB architecture reduces peak-to-peak amplitudes by more than 80% w.r.t. the cMPC.

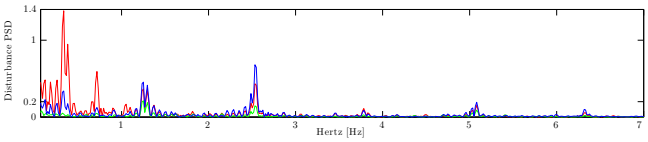

Fig. 6. Power spectral density analysis of the disturbance seen by the WAM robot and induced by beating heart motion. X, Y and Z directions are represented by blue, green and red curves, respectively.

VI. Conclusions

This paper has presented a comparative study of two force control architectures (cMPC and cascade MPC-AOB) dedicated to the beating heart surgery. The cMPC control approach is based on a model of the robotic system to predict future outputs. Due to a pole at the origin the model is unstable, increasing control action dynamics and computational time. Moreover, modeling errors are not compensated, corrupting the plant model. Based on these limitations which greatly affect control tracking performance, we improved the cMPC approach by introducing a Kalman active observer (AOB) in the control architecture. The MPC drives the AOB for autonomous heart motion compensation. This cascade MPC-AOB controller has two loops. The inner loop imposes a desired and stable closed loop dynamics, based on non-linear feedback linearization, augmented state feedback, and stochastic design. The outer loop generates force references for AOB control by predicting applied forces in a finite time horizon. Heart motion compensation capabilities have been evaluated for both controllers. A 3-DoF Heartbox robot equipped with an *ex-vivo* heart generates 3D heart motions, which have been recorded during *in vivo* experiments on a pig's heart. A 4-DoF WAM robot with a force sensor at the tip applies controlled surgical forces on the moving heart. As expected, cMPC control tracking is affected by the unstable plant. Residual peak-to-peak forces are bigger than 2 [N], which is not adequate for beating heart surgery. However, introducing an inner AOB loop to stabilize the system plant, tracking performance greatly improved. The cascade MPC-AOB architecture control showed high quality compensation capabilities. Residual peak-to-peak forces smaller than 0.5 [N] have been attained without knowing *a priori* complex and chaotic heart motions.

Acknowledgments

This work was supported in part by the Portuguese Science and Technology Foundation projects PTDC/EEA-CRO/110008/2009.

References

[1] V. Falk, "Manual control and tracking-a human factor analysis relevant for beating heart surgery," *The Annals of Thoracic Surgery*, vol. 74, pp. 624–628, 2002.

[2] S. Jacobs, D. Holzhey, F. W. Mohr, and V. Falk, "Limitations for manual and telemanipulator-assisted motion tracking and dexterity for endoscopic surgery," in *Proc. of the Int. Conf. on Computer Assisted Radiology and Surgery (CARS)*, June 2003, pp. 673–677.

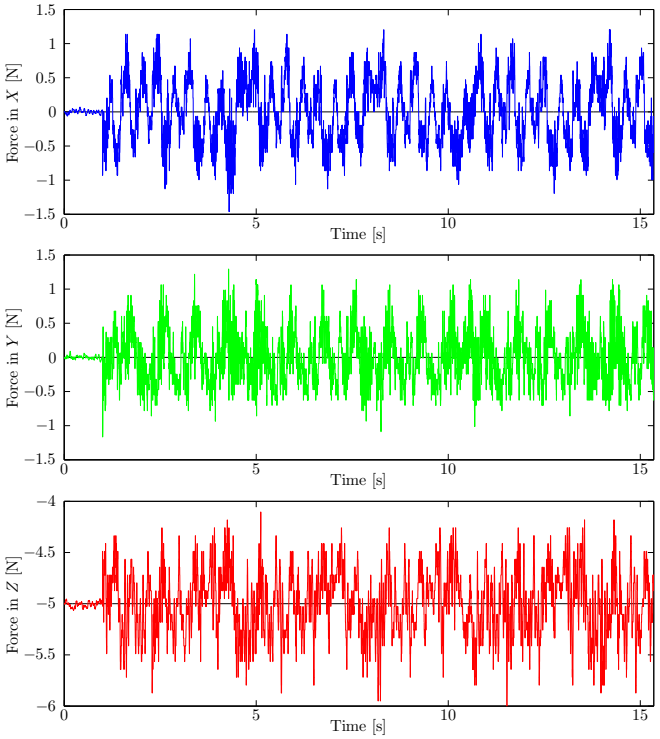

Fig. 7. cMPC experimental results for 3D motion compensation. Applied forces along X, Y and Z directions on an moving heart for constant desired forces (0 [N] for X and Y axis and -5 [N] for Z axis). Physiological motions are introduced at 1 [s] along three directions.

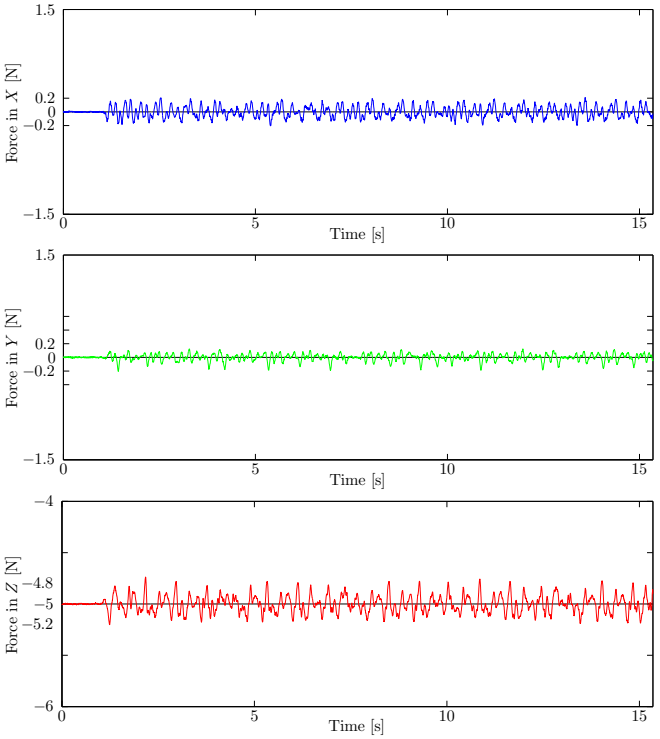

Fig. 8. Cascade MPC-AOB experimental results for 3D motion compensation. Applied forces along X, Y and Z directions on an moving heart for constant desired forces (0 [N] for X and Y axis and -5 [N] for Z axis). Physiological motions are introduced at 1 [s] along three directions.

[3] M. F. Newman, J. L. Kirchner, B. Phillips-Bute, V. Gaver, H. Grocott, and R. H. Jones, "Longitudinal assessment of neurocognitive function after coronary-artery bypass surgery," *The New England Journal of Medicine*, vol. 344, no. 6, pp. 395–402, February 1991.

[4] V. Falk, A. Diegler, T. Walther, R. Autschbach, and F. Mohr, "Developments in robotic cardiac surgery," *Current Opinion in Cardiology*, vol. 15, pp. 378–387, 2000.

[5] R. Dzwonczyk, C. del Rio, B. Sun, R. Michler, and M. Howie, "Devices used to expose the posterior coronary artery in opcabg surgery may cause ischemia," in *Proceedings of the 31th Bioengineering conference*, vol. 2-3, 2005, pp. 148–149.

[6] A. L. Trejos, S. E. Sulcudean, F. Sassani, and S. Lichtenstein, "On the feasibility of a moving support for surgery on the beating heart," in *Proc. of the Second Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. London, UK: Springer-Verlag, 1999, pp. 1088–1097.

[7] W. Bachta, P. Renaud, E. Laroche, A. Forgione, and J. Gangloff, "Cardiolock: an active cardiac stabilizer, first in vivo experiments using a new robotized device," *J. of Computer Aided Surgery*, vol. 13, no. 5, pp. 243–254, September 2008.

[8] R. Richa, A. P. L. Bó, and P. Poignet, "Beating heart motion prediction for robust visual tracking," in *Proceedings of IEEE Conference on Robotics and Automation (ICRA '10)*, Anchorage, USA, 2010, pp. 4579–4584.

[9] M. Kitagawa, A. Okamura, B. Bethea, V. Gott, and W. Baumgartner, "Analysis of suture manipulation forces for teleoperation with force feedback," in *Proc. of the Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, Tokyo, Japan, 2002, pp. 155–162.

[10] B. Cagneau, N. Zemiti, D. Bellot, and G. Morel, "Physiological motion compensation in robotized surgery using force feedback control," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, Roma, Italy, April 2007, pp. 1881–1886.

[11] Y. Nakamura, K. Kishi, and H. Kawakami, "Heatbeat synchronization for robotic cardiac surgery," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, vol. 2, Seoul, Korea, May 2001, pp. 2014–2019.

[12] S. Kesner and R. Howe, "Force control of flexible catheter robots for beating heart surgery," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, Shanghai, China, May 2011, pp. 1589 –1594.

[13] M. Dominici, R. Cortesão, and C. Sousa, "Heart motion compensation for robotic-assisted surgery predictive approach vs. active observer," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, Shanghai, China, May 2011.

[14] M. Dominici, P. Poignet, R. Cortesão, E. Dombre, and O. Tempier, "Compensation for 3D physiological motion in robotic-assisted surgery using a predictive force controller. experimental results," in *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, St Louis, USA, September 2009.

[15] R. Cortesão, "On kalman active observers," *Journal of Intelligent and Robotic Systems*, vol. 48, pp. 131–155, 2007.

[16] K. J. Åström and B. Wittenmark, *Computer controlled systems: theory and design*, 2nd ed. Prentice Hall, 1997.

[17] R. Cortesão, J. Park, and O. Khatib, "Real time adaptive control for haptic telemanipulation with kalman active observers," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 987–999, October 2006.

[18] R. Cortesão, "Kalman techniques for intelligent control systems: Theory and robotic experiments," PhD thesis, University of Coimbra, Coimbra, Portugal, 2003.

[19] J. M. Maciejowski, *Predictive control with constraints*. Prentice Hall, 2002.

[20] D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictive control - part i. the basis algorithm," *Automatica*, vol. 23, no. 2, pp. 137–148, 1987.

[21] ——, "Generalized predictive control - part ii. extensions and interpretations," *Automatica*, vol. 23, no. 2, pp. 148–160, 1987.

[22] M. Sauve, A. Noce, P. Poignet, J. Triboulet, and E. Dombre, "Three-dimensional heart motion estimation using endoscopic monocular vision system: From artificial landmarks to texture analysis," *Biomedical Signal Processing and Control*, vol. 2, no. 3, pp. 199–207, 2007.