4

Simulation Model for the Dynamics Analysis of a Surgical Assistance Robot

Hans-Christian Schneider and Juergen Wahrburg
University of Siegen, Center for Sensor Systems (ZESS)
Germany

1. Introduction

MODI*CAS* (modular interactive Computer Assisted Surgery) represents an integral solution for the software-based combination of a surgical planning software, an optical localization device and a haptic sensor with a mechatronic manipulator in order to support surgical interventions. One key feature of the integral system is to accurately and precisely align any surgical instrument, according to the preoperative planning, in relation to the bony structure of the patient and to intraoperatively ensure the alignment to remain constant all the time. This is made possible due to the automatically controlled tracking of small patient movements in real time. As a result of developed calibration algorithms, the stationary precision and accuracy of the whole system is mainly defined by the measurement characteristics of the applied localization device. Moreover, the actual exploratory focus lies on the enhancement of the dynamic behaviour, especially on the reduction of the dynamic tracking error without concurrently degrading the stationary properties. The following chapter describes the development and use of an offline simulation environment for the analysis and the enhancement of the MODI*CAS* patient tracking system dynamics. At first, the functional principle of the tracking procedure is discussed. Furthermore, the physical modelling of all relevant system characteristics and the identification of the system parameters are described. Additionally, the model behaviour is verified against measurements from the real surgical assistance system. It is shown, that the offline model properly simulates the behaviour of the real system. As an example of use, a comparison of three tracking control strategies is shown on the basis of the developed and identified model. In the future, further simulations will be performed, in order to understand how various system parameters like lags, measurement noise or calibration errors may influence the overall tracking performance. The results will lead to a conclusion about the actual technical constraints and to an outlook on how such system can be further advanced in the future.

2. The modiCAS surgical assistance robot

The concept of the MODI*CAS* surgical assistance system, as shown in Fig. 1 during a clinical trial, has been already introduced in Castillo Cruces *et al.* (2008). The major goal of its concept is to combine a robot manipulator $\{rbs\}$ with a common surgical navigation or localization system $\{ots\}$ respectively to one integral unit (Fig. 2). To be precise, we use a PA10 Series General Purpose Robot from MITSUBISHI HEAVY INDUSTRIES (MHI) as

Source: Robot Surgery, Book edited by: Seung Hyuk Baik,
ISBN 978-953-7619-77-0, pp. 172, January 2010, INTECH, Croatia, downloaded from SCIYO.COM

manipulator and a NDI POLARIS P4 Optical Tracking System $\{ots\}$ as localizer. This integrated device helps the surgeon to accurately and precisely align any surgical instrument $\{ttp\}$ relatively to the patient's anatomy $\{arb\}$, exactly as defined in the computer assisted preoperative planning.

Fig. 1. Clinical tria utilizing the MODI*CAS* assistance robot for total hip arthroplasty implantation

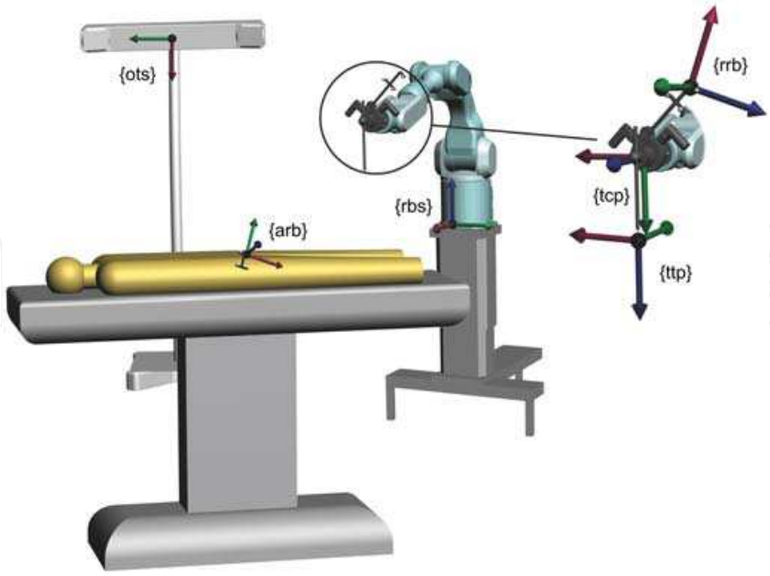


Fig. 2. Coordinate systems – {ots} localizer (optical tracking system), {arb} patient (aim reference body), {rrb} (robot reference body) {rbs} robot base, {tcp} robot wrist (tool center point), {ttp} tool tip

Utilizing a robot manipulator for such positioning tasks offers significant advantages in contrast to pure navigation systems. First, a robot manipulator guided by a precise localization system can position any surgical instrument with a very high precision for a long period of time, without tremor, exhaustion or the possibility of slipping. Second, the surgeon, who is released from the monotonous but straining positioning task, can fully concentrate on the main focus of the intervention, for instance what force he applies to the bony structure when he mills or drills using any wrist-mounted, manually controlled instrument.

One key feature of the MODI*CAS* assistance system is that it does not behave fully autonomously but highly interactively in order to cooperatively assist the surgeon. Therefore, much development work has been concentrated on the cooperative haptic interaction interface, as described in Castillo Cruces *et al.* (2008).

One further research and development goal is to make a rigid fixation of the patient unnecessary by integrating an online tracking function that automatically updates the pose of the aligned instrument in real time if the patient moves. Reducing the dynamic error without degrading the stationary precision of such tracking functionality is a challenging task. In practice, the reachable dynamics and precision are bounded by technical constraints of the system components. Thus, the robot control must be carefully adopted to those system parameters. A reliable simulation model of the whole tracking procedure will be helpful to get a better understanding of the patient tracking principle and the influence of various system parameters on the tracking quality. The development of such a model and the identification of its dynamic parameters, as well as one example of use, will be the focus of this article.

3. Real time tracking of patient's movements by the robot

Due to the fact that the MODI*CAS* patient tracking procedure is carried out by the use of an optical tracking system, it can be characterized as a so called *'visual servoing system'*, like already described in Weiss *et al.* (1987). In the past, *visual servoing* approaches have been categorized in detail, depending on the type of e.g. camera or control principle. A generalized overview is given in Kragic & Christensen (2002). This section will illustrate how the MODI*CAS* patient tracking procedure works, by categorizing it and establishing its kind of implementation.

The PA10 robot arm from MHI is shipped as a modular system that allows three different ways of interfacing. The easiest way is to use a dedicated MOTION CONTROL BOARD (MHI MCB) that carries out the entire basic functionality which is typically provided with common industrial robots e.g. like forward and inverse kinematics calculations, control in cartesian or joint space and trajectory path planning. The MHI MCB was utilized within the first generation of the MODI*CAS* assistance system in order to fulfill the general proof of principle of the overall MODI*CAS* concept. The experiences with that first prototype emphasized the necessity of interfacing the robot on a lower level in order to implement new desired features. For instance, such features are a singularity robust haptic interface, virtual motion constraints, calibrated kinematics or in general the possibility to influence the dynamic behaviour of the controlled robot in a more direct way. For such purpose, the robot can be interfaced through direct joint control. Either in torque mode, where control commands are directly interpreted by the robot as joint torque commands and straightly turned into motor currents. Or in velocity mode, where the control commands are

interpreted as velocity commands and the tracking of the velocity command trajectories is carried out jointwise by internal PI Controllers per each servo. Even though promising approaches are existing in literature concerning model-based computed torque control of a PA10 robot (Kennedy & Desai (2004), Bompos *et al.* (2007)), one global development strategy for the actual MODI*CAS* prototype was fixed to retain a cascade control structure for joint angle control, where the servodriver-internal PI velocity controllers robustly compensate disturbances or physical effects like e.g. gravity, coriolis force and friction, respectively.

If the robot kinematics, describing the geometric relation between the joint angles and the robot wrist pose, are exactly calibrated and further the geometric relation between the base coordinate frames of camera $\{ots\}$ and robot $\{rbs\}$ is also exactly known and rigidly fixed, then it would be possible to omit the optical tracking of the robot wrist $\{rrb\}$. Only the optical tracking of the patient's pose $\{arb\}$ would be necessary in order to generate a corresponding joint angle command vector $\vec{q}_c = f\left(T_{arb}^{ots}\right)$ for the robot controller in order to track patient's movements. Such assembly is defined in Kragic & Christensen (2002) as *'endpoint open loop'* configuration.

Certainly, common uncalibrated robot manipulators have significant kinematic errors. Due to manufacturing tolerances, the real kinematics differ from their nominal model. This leads to a reduced positioning precision depending on the dimension of kinematic errors, even if the manipulator has a good repeatability (Bruyninckx & Shutter (2001)). Further it is extremely challenging to permanently guarantee an exactly fixed geometric relationship between the base coordinate systems of the robot and the camera, if the camera acts from an observer perspective (outside-in or stand-alone, as defined in Kragic & Christensen (2002), respectively). Therefore, within the MODI*CAS* tracking procedure, the robot wrist is optically tracked as well. That facilitates the compensation of kinematic errors or changes in the geometric relationship between the camera's and the robot's base frame. Due to the additional optical tracking of the robot wrist or end effector respectively, such assembly is defined as *'end point closed loop'* configuration.

In principle, due to the optically closed loop, an underlying feedback from the robot's joint encoders is not essentially required to perform visual servoing. Omitting such joint encoder feedback would lead to a so called *'direct visual servoing'* system, where the dynamic control of the robot is carried out directly through the optical feedback loop.

Due to the fact that typical surgical optical tracking systems like the NDI-POLARIS have a relatively low bandwidth in contrast to common robot joint encoders, it is reasonable to use a so called *look and move* approach. Here, the potential of precise but maybe slow optical sensors to compensate kinematic errors is profitably combined with the higher bandwidth of the joint encoders by retaining the joint encoder feedback.

By the reason that surgical optical tracking systems commonly deliver full position and orientation of all tracked elements in the three-dimensional space, the robot control can be performed *'position based'*. The opposite of *position based* is *'image based'*, where the control law is directly based on raw image features instead of fully determined 3D pose data.

Finally, due to the utilization of a stereo vision system which is not rigidly fixed to the robot wrist (like inside-out or eye in hand systems, as defined in Kragic & Christensen (2002), respectively), but acts from an observer perspective (Fig. 2), we can classify the MODI*CAS* patient tracking approach as *position based dynamic look and move using outside-in stereovision in endpoint closed loop configuration*.

A block diagram of the MODI*CAS* patient tracking principle is illustrated in Fig. 3. Here, the robot is dynamically controlled in joint space. Due to that it is interfaced in velocity mode, all joints are velocity-controlled by their servodriver-internal PI controllers that cannot be modified. However, the overlying joint angle control loops may be customized in order to adapt the dynamic behaviour at the best to the desired patient tracking functionality. Available input signals for implementing any desired joint angle controllers are the joint angle command vector $\vec{q}_c$, the joint angle feedback vector $\vec{q}$ and the joint velocity feedback vector $\dot{\vec{q}}$.
In order to follow patient's movements, the control input for the decentralized joint control of the robot must represent the joint angle vector

$$\vec{q}_c = IK\left(T_{arb}^{rbs} \times T_{tcp}^{arb}\right), \tag{1}$$

where the inverse kinematics *IK* give the joint angle vector $\vec{q}_c$ that corresponds to that robot arm configuration T_{tcp}^{rbs} where the robot wrist strikes a desired pose T_{tcp}^{arb} relatively to the patient's reference pose T_{arb}^{rbs}.
If the determination of the patient's pose is carried out through an *outside-in* localization system in *endpoint closed loop configuration*, then the tracking algorithm results in a direct geometric coordinate transformation equation such that

$$\vec{q}_c = IK\left(\underbrace{T_{tcp}^{rbs}}_{FK(\vec{q})} \times \underbrace{\underbrace{\left[T_{rrb}^{ots} \times T_{tcp}^{rrb}\right]}_{actual\ T_{tcp}^{ots}}{}^{-1} \times \underbrace{T_{arb}^{ots} \times T_{tcp}^{arb}}_{desired\ T_{tcp}^{ots}}}_{\underline{E}}\right), \tag{2}$$

where *FK* are the forward kinematics calculated on the basis of the actual robot joint angle measurements $\vec{q}$; T_{tcp}^{rrb} is a constant matrix derived from the robot to localizer calibration; T_{tcp}^{arb} is the desired constant pose or trajectory of the robot wrist relatively to the patient reference frame; T_{rrb}^{ots}, T_{arb}^{ots} are the frames of the optically measured reference bodies and $\underline{E}$ is the optically determined pose error matrix.
Primally when the optically measured pose of the robot wrist is exactly the same as the desired one relatively to the optically measured pose of the patient, then the equation

$$\underline{E} = \left[T_{rrb}^{ots} \times T_{tcp}^{rrb}\right]^{-1} \times T_{arb}^{ots} \times T_{tcp}^{arb} = \underline{I}, \tag{3}$$

where $\underline{I}$ is the identity matrix, is fulfilled, such that the system is compensated and the actually measured joint angles are directly fed through as setpoint values

$$\vec{q}_c = IK\left(FK\left(\vec{q}\right) \times \underline{I}\right). \tag{4}$$

However, if any pose error $\underline{E}$ occurs due to displacement of the patient $\{arb\}$, the localizer $\{ots\}$ or the robot base $\{rbs\}$ or due to kinematic uncertainties, the tracking algorithm geometrically calculates the desired robot wrist pose that is needed to fulfill equation 3. The dynamic compensation rather takes place in joint space and is carried out through the joint angle controllers. Thus, a manipulation of the tracking dynamics is exclusively carried out by adapting these joint angle controllers.

The functional separation of the tracking procedure into a geometrically setpoint determination and into a dynamic control exclusively in joint space facilitates the use of classical approaches from control theory for each joint in order to design a fast and robust tracking controller.

4. Simulation model describing the real time tracking procedure

Regarding the objective of tuning the MODI*CAS* tracking procedure at the best, a reliable model-based environment, that exactly represents the real world, facilitates watching process variables or changing parameters that are not observable or manipulable respectively within the real system, yet. For instance, such model-based environment allows experiments e.g. regarding the questions, how far the tracking procedure may be improved with upcoming faster localization systems that are not available yet, or how far miscalibration or kinematic errors affect the system stability without the presence of any accidental risk. The following sections illustrate the dynamic model that has been developed with the objectives to perform a detailled offline analysis of the MODI*CAS* patient tracking procedure and to find the best system tuning in view of all current and persisting technical constraints.

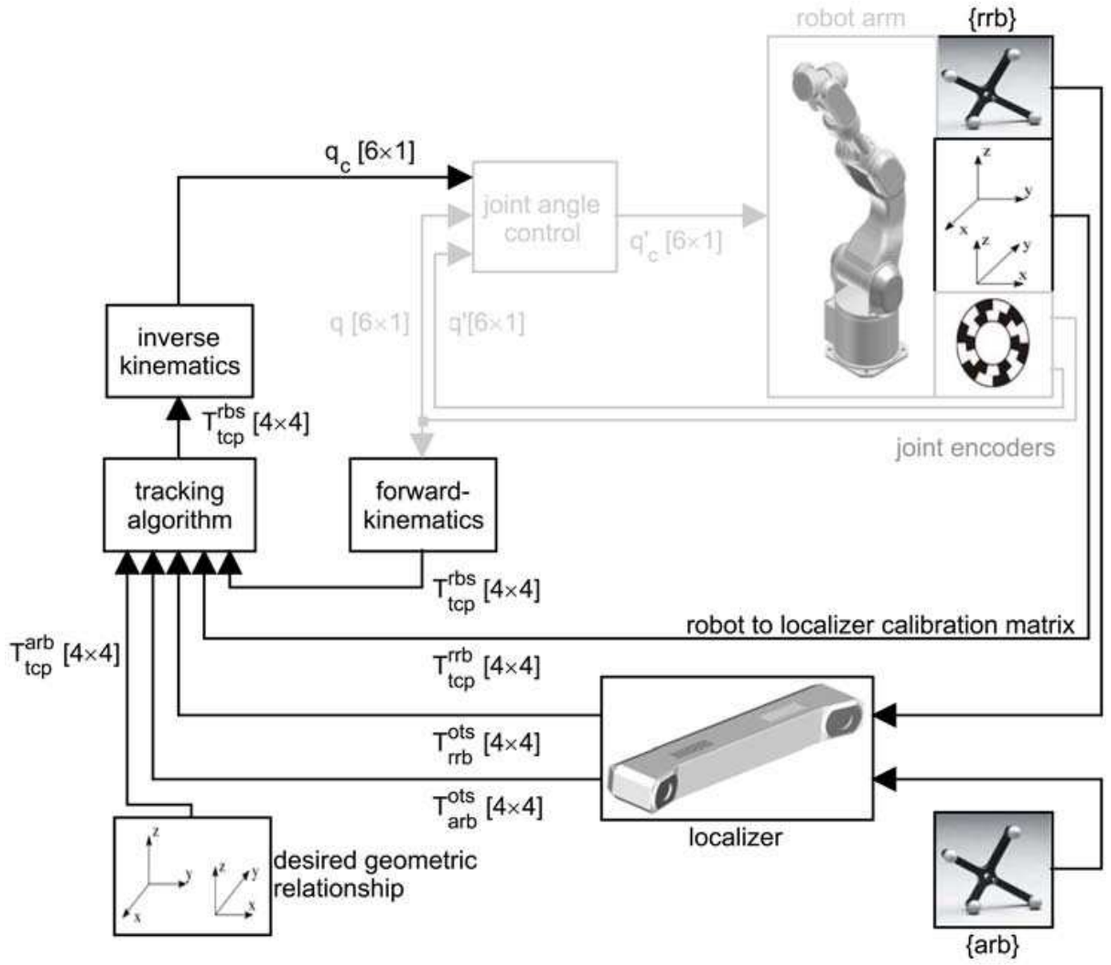


Fig. 3. Block diagram of the MODI*CAS* real time patient tracking procedure

4.1 Global model structure

The global structure of the offline model is directly derived from the block diagram in Fig. 3 which describes the tracking procedure. The forward and inverse kinematics as well as the tracking algorithm itself are straightly copied from the real control software that previously has been implemented within the MODI*CAS* real time control development environment. This environment has been introduced in Schneider & Wahrburg (2008).

4.2 Robot model

The model of the robot arm itself consists of a stiff kinematic model and six structurally identical dynamic joint models with individual joint specific parameters.
The kinematic model, as well as the nominal model in the robot controller, are based on the so called *321-kinematic structure* which is further described in Bruyninckx & Shutter (2001). Due to some simplifying conventions concerning the kinematic structure, the *321-kinematics* model saves some geometric parameters and thus significant computational load in contrast to a common *Denavit-Hartenberg* model. As a result of that simplification, a full identification and thus an exact simulation of the real kinematic errors will not be possible as long as the 321- kinematics model is used. For that purpose, a full implementation of the *Denavit-Hartenberg* convention would be necessary. However, for simple experiments on how kinematic uncertainties affect the behaviour of the tracking procedure, it is sufficient to merely simulate joint angle offsets as well as link length errors (Fig. 4).

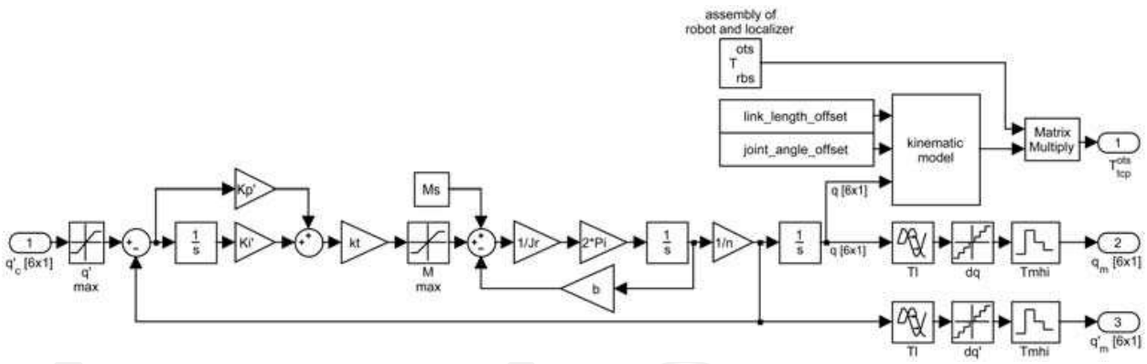


Fig. 4. Robot model - kinematics and joint servo dynamics

Regarding the dynamics, any disturbances or physical effects like e.g. gravity, that act on the gear sides of the real robot joints, are strongly reduced at the motor sides through high gear ratios and therefore relatively small in relation to the inertias of the joint servo rotors. Further, due to the fact that, within the MODI*CAS* system, the PA10 robot is interfaced in velocity mode, such effects are quickly compensated through the servodriver-internal PI velocity controllers. Therefore, it is adequate to model every joint drive as a simple PI-controlled dc-motor as it is illustrated in Fig. 4, in order to authentically simulate the dominant dynamic behaviour of the robot arm in velocity mode. All joint model parameters are listed in Tab. 1.
Those parameters that cannot be determined straightly from available technical data sheets of the robot, are identified by fitting the velocity step response of every joint model into its corresponding measurement from the real system. The estimation of the unknown parameters is carried out through `fmincon()` from the MATLAB OPTIMIZATION TOOLBOX

$\dot{q}_{max}$	maximum velocity command input	$[\frac{rad}{s}]$
M_{max}	maximum torque	$[Nm]$
K'_p	proportional gain of velocity controller	$[\frac{As}{rad}]$
K'_i	integral gain of velocity controller	$[\frac{As}{rad}]$
k_t	motor constant	$[\frac{Nm}{A}]$
J_r	cummulative moment of inertia in the powertrain	$[kg \cdot m^2]$
T_l	joint encoder output lag	$[s]$
dq	joint encoder angle resolution	$[\frac{rad}{digit}]$
$d\dot{q}$	joint encoder angular velocity resolution	$[\frac{rad}{s \cdot digit}]$
T_{mhi}	sample time of the robot interface in velocity mode	$[s]$
b	viscous friction	$[\frac{Nms}{rad}]$
M_s	disturbance torque	$[Nm]$
n	gear ratio	

Tab. 1. parameters of one joint model

which manipulates all unknown parameters within user defined constraints and performs a simulation per each parameter set, until a quality function, defined as

$$Q = \frac{a_1}{n} \sum_{i=1}^{n} [q_{ms}(i) - q_{sm}(i)]^2 + \frac{a_2}{n} \sum_{i=1}^{n} [\dot{q}_{ms}(i) - \dot{q}_{sm}(i)]^2 , \tag{5}$$

reaches a minimum, where q_{ms} is the measured and q_{sm} the simulated joint angle, $\dot{q}_{ms}$ is the measured and $\dot{q}_{sm}$ the simulated angular velocity and a_1, a_2 are manipulable weighting gains.

Fig. 5 shows the result of the described identification procedure, exemplarily for the first shoulder joint of the robot (*S*1). Due to the simple structure of the joint model, the torque curve is strongly idealized. However, the model reproduces the angle and angular velocity trajectories of the real joint drive very well, if stimulated with the same velocity command like the real one. In order to check if these characteristics are reproducible over the full workspace of the robot, independently from payload, robot arm configuration or input signals, several verification tests were done. Exemplarily, Fig. 6 shows a verification result where, due to a changed robot arm configuration (see Fig. 7), a lower moment of inertia acts on the joint *S*1 and further the velocity command is 0.1 $\frac{rad}{s}$ higher than during the identification process. The simulation is carried out using exactly the same parameters as in the experiment illustrated in Fig. 5. Even though the real torque characteristics differ between the two experiments due to a changed moment of inertia acting on joint *S*1, the simulated angle and angular velocity trajectories always exactly represent the corresponding measurements from the real joint drive. Thus, the developed dynamic joint models are fully adequate to simulate the dynamic behaviour of the robot within the tracking procedure.

4.3 Localizer model

At the actual state of development, the localizer model merely simulates the time performance of the NDI-POLARIS-System and normally distributed spatial measurement noise, whereas the sampling time ΔT_{ots} as well as the measurement lag t_{lots} can be globally changed and the standard deviation for each component of a measured pose (x,y,z,α,β,γ) can

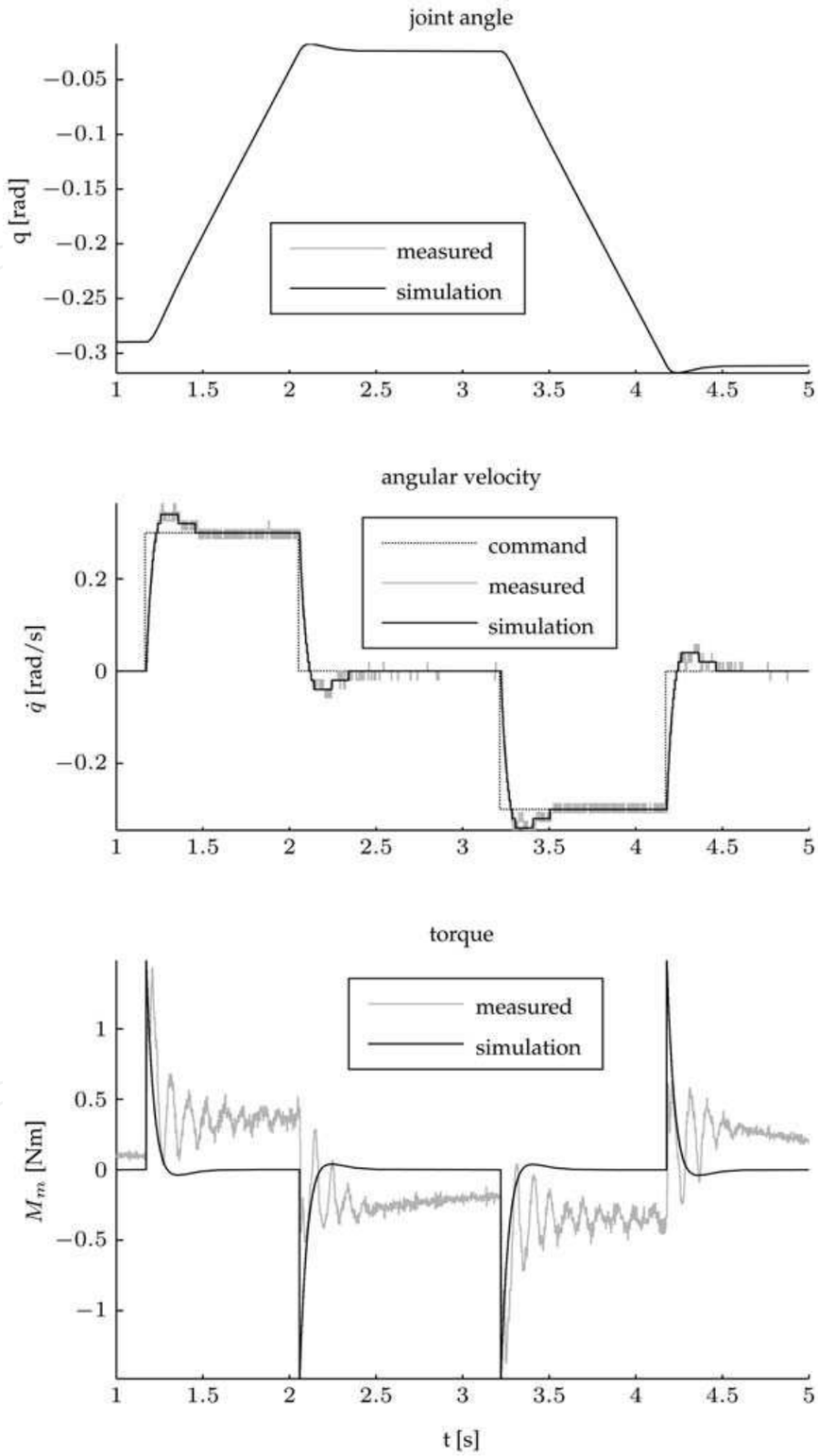


Fig. 5. Simulation results using the identified servo model compared to real measurements (dataset for identification), exemplary for the first shoulder joint ($S1$)

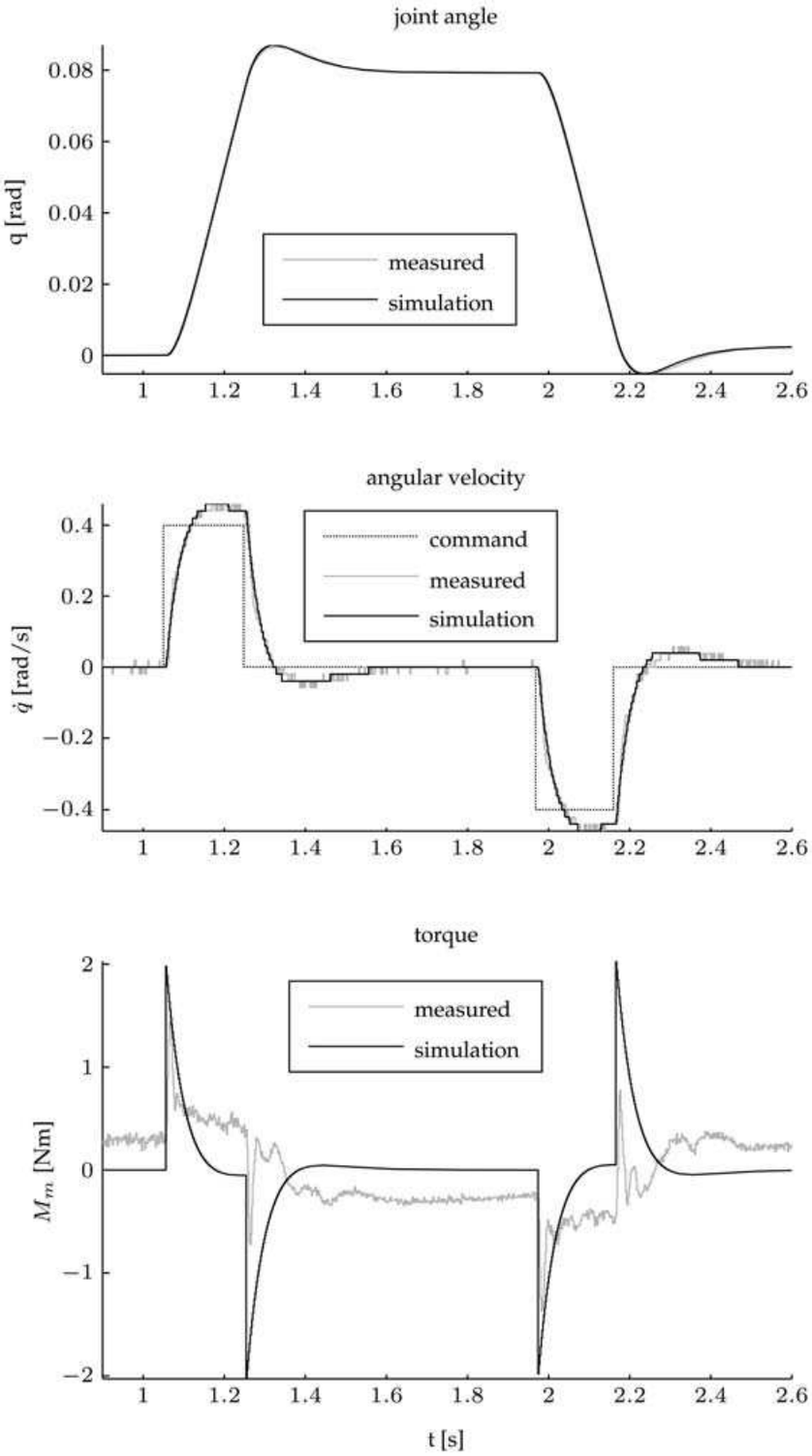


Fig. 6. Simulation results using the identified servo model compared to real measurements for the first shoulder joint (*S*1) in one exemplary scenario different to the identification scenario

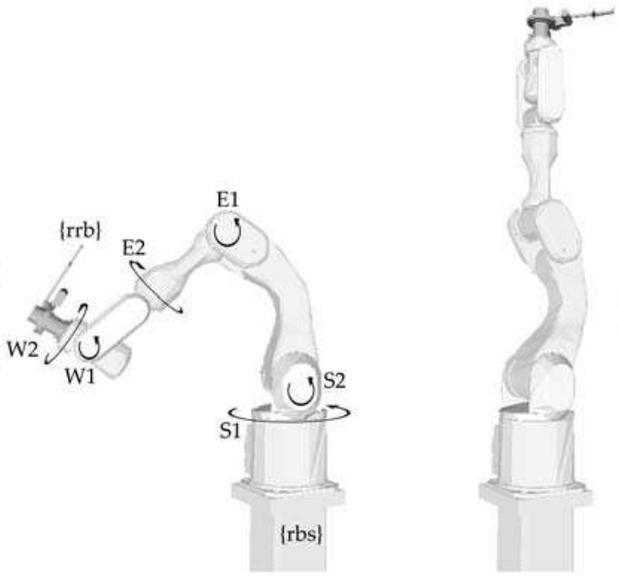


Fig. 7. Differing poses for robot dynamics identification (left) and exemplary verification (right)

be individually manipulated for each simulated reference body. Further, miscalibration between the robot wrist $\{tcp\}$ and its optical reference body $\{rrb\}$ can be simulated through multiplying a corresponding error transformation matrix T_ϵ.

The localizer model is actually kept that simple because the current focus lies in exploring how the time performance of any (replaceable) localizer influences the overall tracking behaviour. If a strongly detailed measurement error model of the NDI-POLARIS with its anisotropic measurement characteristics will be desired, mathematical models like e.g. from Wiles *et al.* (2008), an extension of Fitzpatrick *et al.* (1998), can be integrated into the dynamic model of the MODI*CAS* patient tracking procedure in the future.

4.4 Model verification

The full dynamic model that is described above, consisting of the robot model, localizer model, kinematics and tracking algorithms, is verified against measurements from the real MODI*CAS* assistance system while tracking random patient movements. In order to better enable the recognition of dynamic transients in the laboratory, the applied patient motion is much faster than typically expected during any surgical intervention. The results of one verification experiment are presented in Fig. 8 as cartesian trajectories. The corresponding time trajectories of the robot joint angles are further illustrated in Fig. 9. As it can be seen in Fig. 9, especially in the plots for the joints *E*1 and *W*1, there are noticeable differences between the simulated and the measured time trajectories of the joint angles. What firstly seems to be a weak point of modelling, is a valuable feature of the tracking principle from equation (3). Not only does the observed level deviation occur in the joint responses, but it also occurs in the joint angle command trajectories. The reason for that phenomena is that, for the exemplarily presented simulation, no kinematic error has been considered in the model. While the real uncalibrated robot has significant kinematic errors, in the simulation all parameters for link length errors and joint offsets (Fig. 4) were set to zero. Although the real robot has significant kinematic errors, the tracking algorithm within the real system

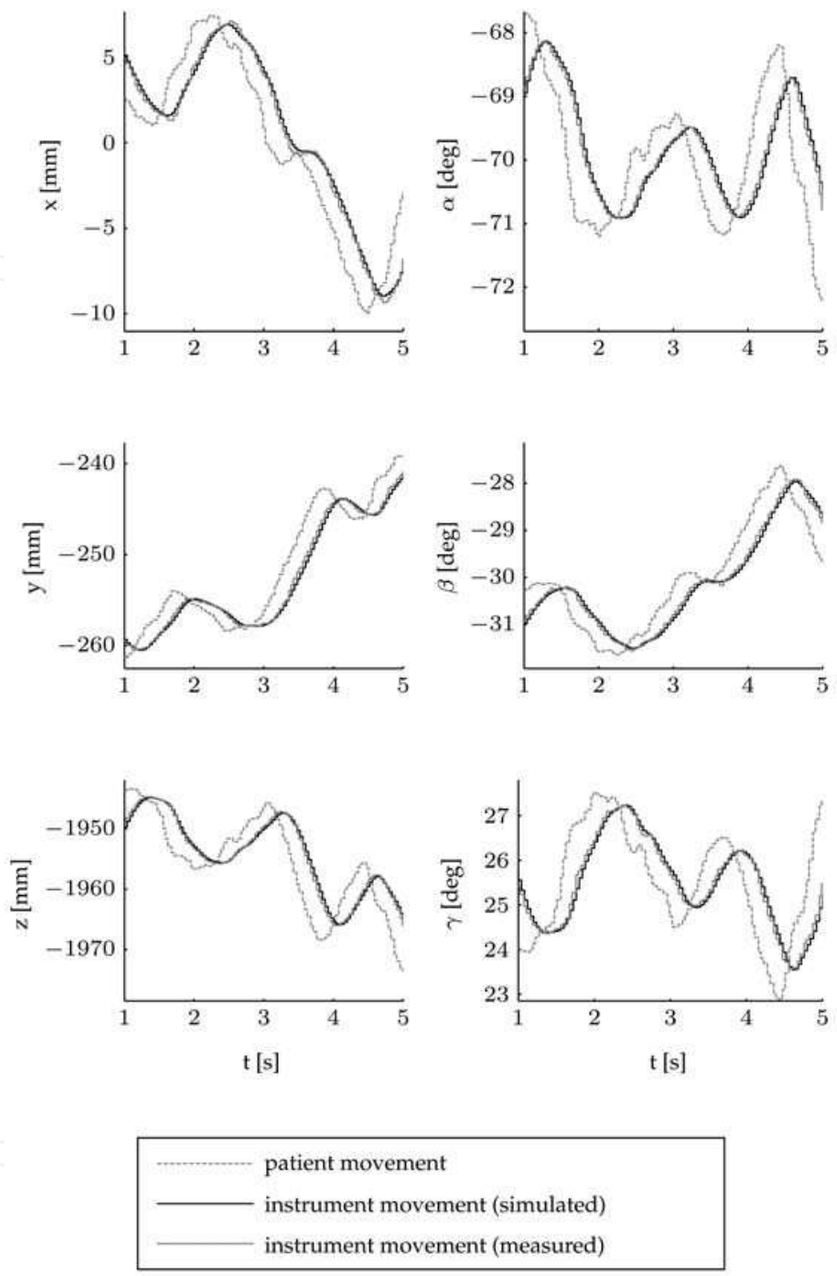


Fig. 8. Verification of the overall tracking procedure model by comparing the model output signals to real measurements (cartesian time trajectories of robot wrist pose)

adjusts the joint angle trajectory commands such that the cartesian trajectories match those of a kinematically precise robot (as simulated in Fig. 8). Accordingly, there will not remain any kinematically caused deviation between the actual and desired pose of the robot wrist in steady state. All in all, Fig. 8 clearly indicates that the simulation of the MODICAS patient tracking function represents the real system behaviour very well and the developed simple model is fully adequate for further investigations, in presence of a joint velocity interfaced robot.

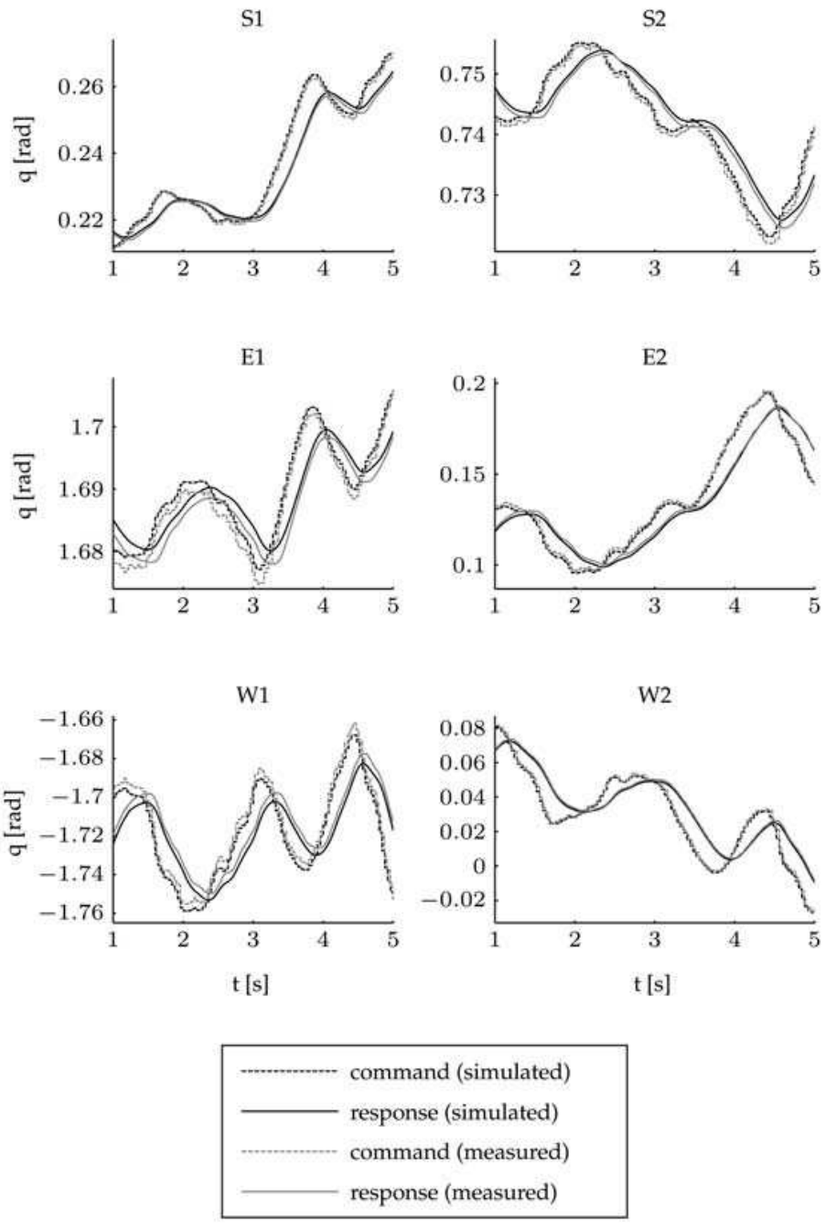


Fig. 9. Verification of the overall tracking procedure model by comparing the model output signals to real measurements (time trajectories of robot joint angles)

5. Application example: model-based controller design

One of the first simulation experiments that have been performed using the novel offline model environment was aimed to compare different control strategies, especially adopted to the time performance of the NDI-POLARIS, at first under the consumption of zero measurement noise. A more general investigation on different control structures within (*image based*) visual closed loop systems has been already presented in Corke & Good (1996). However, our investigation is specially aimed at finding the best possible control strategy

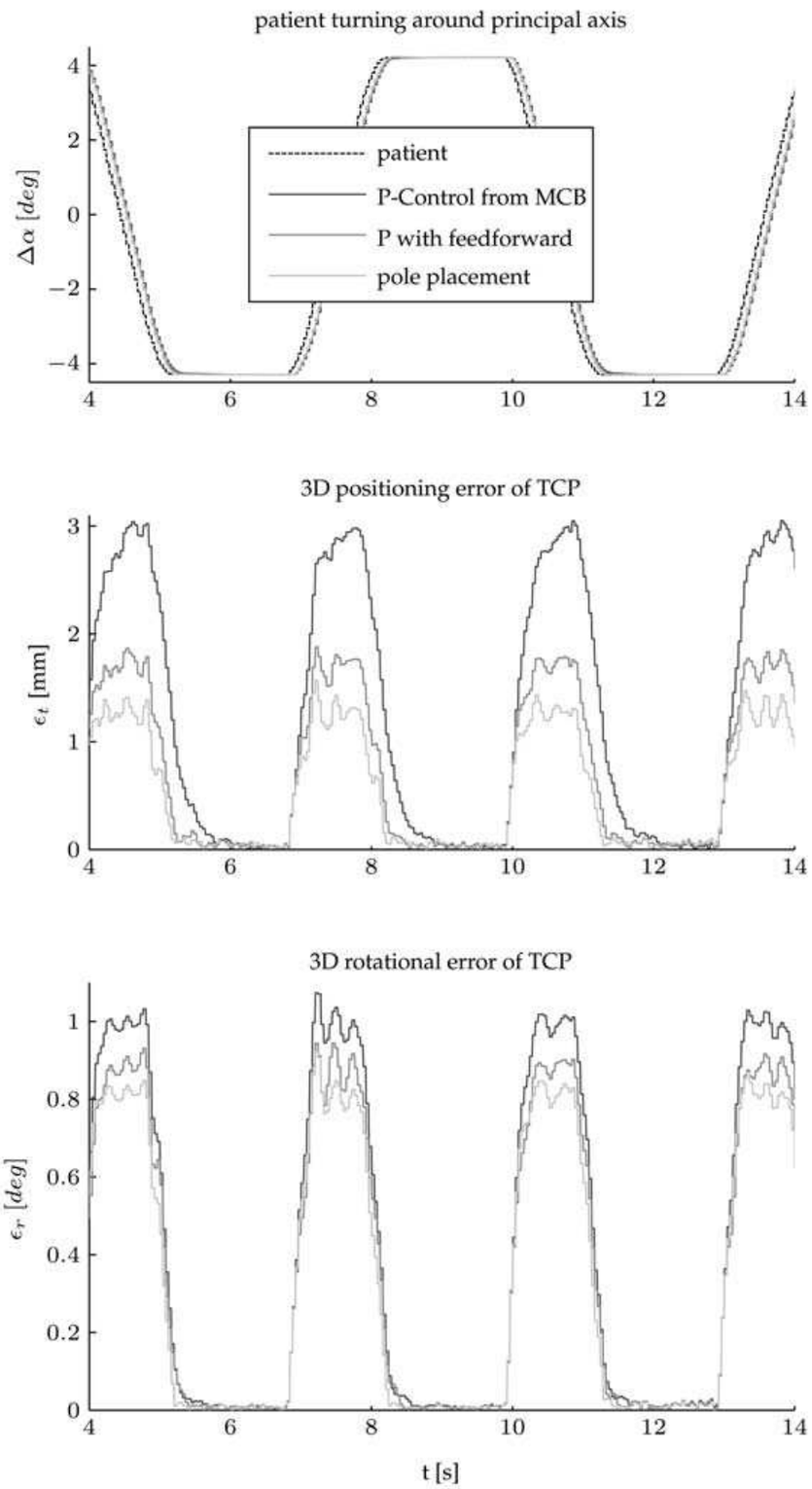


Fig. 10. Comparison of three different control strategies - 1. original proportional controller, 2. proportional with feedforward, 3. pole placement controller

for the MODI*CAS* patient tracking procedure with its functional principle like shown in Fig. 3. As communicated through the manufacturer, only a proportional controller per each joint is generally implemented into the original MHI MCB. With the help of the model-based simulation environment we found out that, regarding the patient tracking function, a specially tuned proportional feedforward controller enhances the overall tracking behaviour, although the feedforward velocity signal, that can be derived from the NDI-POLARIS through a simple differentiation, is poor due to the relatively low sample rate in relation to the time response of the robot.

A further enhancement has been carried out by the use of a pole placement controller, especially developed and adopted to the time performance of the NDI-POLARIS. The results of the two enhanced controllers may be compared to the original one from the MHI MCB by reckoning Fig. 10. In the experiment, a patient dummy is rotated around a defined axis that is assumed to be the principal axis of the patient. The measurement of the patient dummy's reference body is fed into the offline model, where the patient tracking procedure is simulated three times using three different controllers. First, the proportional controller with the original controller gains from the MHI MCB, second the specially tuned proportional feedforward controller and third, the specially developed pole placement controller. As the plots show, in the exemplary tracking experiment, the maximum 3D positioning error is reduced when using the proportional feedforward controller and further reduced when using the pole placement controller. Upcoming experiments will further show how far those controllers can be sufficiently be utilized in the presence of measurement noise or how to then find the best compromise between fast dynamics and high accuracy as well as precision in steady state, respectively.

6. Conclusion

The investigation that is described in this chapter has derived an offline model that simulates the system dynamics of the real MODI*CAS* patient tracking procedure very well, independently from the operating point of the system. The model enables the developer to better understand the functional principle of the tracking procedure and to perform a specific tuning of its parameters in order to increase its overall dynamic performance. One model-based experiment has already delivered an improvement of the tracking control strategy. In the future, further experiments will show how the improvement of the localizer device, especially by means of noise reduction and a faster data aqcuistion, can enhance the overall dynamic performance of the tracking procedure.

7. References

Bompos, N., Artemiadis, P., Oikonomopoulos, A., & Kyriakopoulos, K. (2007). Modeling, full identification and control of the mitsubishi PA-10 robot arm. In *2007 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. 1–6.

Bruyninckx, H. & Shutter, J. (2001). Introduction to intelligent robotics. Tech. rep., Katholieke Universiteit de Leuven.

Castillo Cruces, R., Schneider, H., & Wahrburg, J. (2008). Cooperative robotic system to support surgical interventions. In *Medical Robotics*, Vienna, Austria: I-Tech Education and Publishing, chap. 35. 481–490.

Corke, P. & Good, M. (1996). Dynamic effects in visual closed-loop systems. *IEEE Transactions on Robotics and Automation*, *12*(5), 671–683.

Fitzpatrick, J., West, J., & Maurer, C. (1998). Predicting error in rigid-body point-based registration. *IEEE Transactions on medical imaging*, *17*(5), 694–702.

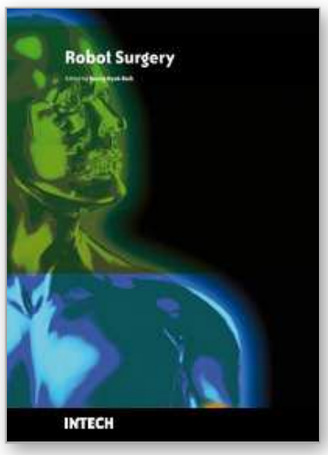
Kennedy, C. & Desai, J. (2004). Model-based control of the Mitsubishi PA-10 robot arm: application to robot-assisted surgery. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA'04)*.. New Orleans, LA, USA, vol. 3, 2523–2528.

Kragic, D. & Christensen, H. (2002). Survey on visual servoing for manipulation. Tech. rep., School of Computer Science and Communication, Centre for Autonomous Systems, Numerical Analysis and Computer Science, Stockholm, Sweden.

Schneider, H. & Wahrburg, J. (2008). A Novel Real Time Core for Dynamics and Safety Enhancement of a Navigated Surgical Assistance Robot. *at-Automatisierungstechnik*, *56*(9), 483–493.

Weiss, L., Sanderson, A., & Neuman, C. (1987). Dynamic sensor-based control of robots with visual feedback. *IEEE Journal of Robotics and Automation*, *3*(5), 404–417.

Wiles, A., Likholyot, A., Frantz, D., & Peters, T. (2008). A Statistical Model for Point-Based Target Registration ErrorWith Anisotropic Fiducial Localizer Error. *IEEE TRANSACTIONS ON MEDICAL IMAGING*, *27*(3), 378–390.



Robot Surgery

Edited by Seung Hyuk Baik

ISBN 978-953-7619-77-0
Hard cover, 172 pages
Publisher InTech
Published online 01, January, 2010
Published in print edition January, 2010

Robotic surgery is still in the early stages even though robotic assisted surgery is increasing continuously. Thus, exact and careful understanding of robotic surgery is necessary because chaos and confusion exist in the early phase of anything. Especially, the confusion may be increased because the robotic equipment, which is used in surgery, is different from the robotic equipment used in the automobile factory. The robots in the automobile factory just follow a program. However, the robot in surgery has to follow the surgeon's hand motions. I am convinced that this In-Tech Robotic Surgery book will play an essential role in giving some solutions to the chaos and confusion of robotic surgery. The In-Tech Surgery book contains 11 chapters and consists of two main sections. The first section explains general concepts and technological aspects of robotic surgery. The second section explains the details of surgery using a robot for each organ system. I hope that all surgeons who are interested in robotic surgery will find the proper knowledge in this book. Moreover, I hope the book will perform as a basic role to create future prospectives. Unfortunately, this book could not cover all areas of robotic assisted surgery such as robotic assisted gastrectomy and pancreaticoduodenectomy. I expect that future editions will cover many more areas of robotic assisted surgery and it can be facilitated by dedicated readers. Finally, I appreciate all authors who sacrificed their time and effort to write this book. I must thank my wife NaYoung for her support and also acknowledge MiSun Park's efforts in helping to complete the book.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Hans-Christian Schneider and Juergen Wahrburg (2010). Simulation Model for the Dynamics Analysis of a Surgical Assistance Robot, Robot Surgery, Seung Hyuk Baik (Ed.), ISBN: 978-953-7619-77-0, InTech, Available from: http://www.intechopen.com/books/robot-surgery/simulation-model-for-the-dynamics-analysis-of-a-surgical-assistance-robot

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820

Fax: +385 (51) 686 166
www.intechopen.com

Fax: +86-21-62489821

IntechOpen