APPLYING EFFICIENT COMPUTATION OF THE MASS MATRIX FOR DECOUPLING CONTROL OF COMPLEX PARALLEL MANIPULATORS

Houssem Abdellatif * **Bodo Heimann** *
Christian Holz **

* *Hanover Center of Mechatronics, University of Hanover*
** *Institute of Production Engineering and Machine Tools, University of Hannover*

Abstract: Most challenging aspects in the control of complex parallel mechanisms or robotic manipulators is how to deal with the high system nonlinearity and with the coupled dynamics. These aspects can be satisfied by integrating the pose dependent inertia or mass matrix into the control scheme. To satisfy the requirements of real-time application, this paper presents a high efficient method to compute the mass matrix of arbitrarily complex parallel manipulators. Model simplification become dispensable, which leads necessarily to the improvement of control accuracy.

Keywords: Robotic Manipulators, Parallel Manipulators, Nonlinear Dynamics, Decoupling Control, Mass Matrix

1. INTRODUCTION

It is well known that research on control for robotic systems and manipulators has done important steps over the last decades. On this issue a huge amount of literature exists. A very recent and excellent overview is given in (Khalil and Dombre, 2002). Unfortunately there still is a big gap between research and industrial application, where single-joint controllers are yet state of the art, even if it is known in the meantime, that model based feedforward control yields much higher accuracy.

Since (Chae *et al.*, 1986) it was experimentally proven for serial manipulators, that the more their nonlinear dynamics are considered and compensated, the better the control results are. Such feedforward compensation, also called computed-force control, is even more crucial for parallel kinematic manipulators (PKM), since such systems are generally characterized with high nonlinear, coupled and very complex dynamics (Merlet, 2000). It was schown in (Honegger *et al.*, 2000), that simple single-joint controllers cause superproportional errors at velocities over 0.4 ms^{-1}. This is simply intolerable, since PKM are supposed to be advantageous in the range of high dynamics (Grotjahn *et al.*, 2004). Taking account of the nonlinear and coupled dynamics by means of feedforward control yields considerable improvement of accuracy (Denkena *et al.*, 2004; Abdellatif *et al.*, 2004; Honegger *et al.*, 2000). Besides the computation of the pose dependent mass matrix has a key role in axes-decoupling control (Courdourey, 1996; Hesselbach *et al.*, 2004), which was even recognized by the industry. Siemens is making first steps towards the integration of nonlinear dynamics in form of mass matrix computation in standard industrial control systems (Puchtler *et al.*, 2004). Unfortunately the real-time computation of the dynamics for PKM remains challenging, which could explain why the mentioned approaches were validated only on simple structures.

In this paper a precise and computational efficient algorithm for the calculation of the mass matrix of complex PKM is presented. By regarding the manipulator as a set of different kinematic chains, the effort of calculation can be considerably reduced (section 3). The successful evaluation on a complex 6-DOF-Stewart-Gough-platform with 19 bodies is shown in section 4. Finally the application of the mass matrix in decoupling control is demonstrated with experimental results in section 5.

2. USE OF MASS MATRIX FOR DECOUPLING CONTROL

An important aspect to develop control strategies for robotic manipulators is the consideration of nonlinearities. Controller for PKM require therefore more sophistication due to the high mechanical coupling. Multibody systems with closed kinematic chains are characterized by considerable inertia variations effects (Courdourey, 1996; Grotjahn *et al.*, 2004), which have to be involved into control strategies to achieve desired accuracy and high dynamics. The pose dependent mass matrix $\boldsymbol{M}$ of a PKM is the most important factor, characterizing configuration dependent, nonlinear and coupled dynamics. Its consideration can be achieved in schemes of feedforward control (Khalil and Dombre, 2002; Honegger *et al.*, 2000)(see also Fig. 1) or in computed-torque feedback control (Hesselbach *et al.*, 2004).

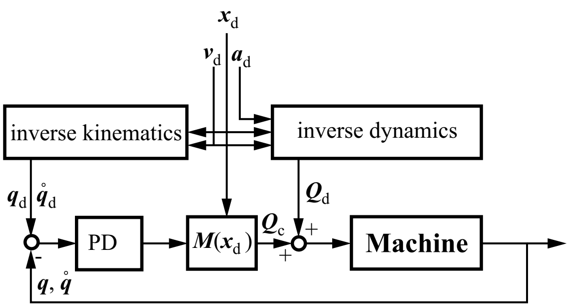

Fig. 1. General control approach for robotic manipulators using feedforward and decoupling control in combination with single joint PD-controller

Considering a robotic manipulator with N_B bodies, the mass matrix can be obtained by means of the Lagrangian formalism:

$$\boldsymbol{M} = \sum_{i=1}^{N_\mathrm{B}} \left[m_i \left(\frac{\partial \bar{\boldsymbol{v}}_i}{\partial \dot{\boldsymbol{\theta}}} \right)^\mathrm{T} \frac{\partial \bar{\boldsymbol{v}}_i}{\partial \dot{\boldsymbol{\theta}}} + \left(\frac{\partial \bar{\boldsymbol{\omega}}_i}{\partial \dot{\boldsymbol{\theta}}} \right)^\mathrm{T} \bar{\boldsymbol{I}}_i \frac{\partial \bar{\boldsymbol{\omega}}_i}{\partial \dot{\boldsymbol{\theta}}} \right]. \tag{1}$$

The minimal coordinates and velocities of the manipulator are denoted in the following by $\boldsymbol{\lambda}$ and $\dot{\boldsymbol{\theta}}$ respectively. $\bar{\boldsymbol{v}}_i$ and $\bar{\boldsymbol{\omega}}_i$ denote the translation and rotational velocities of the center of gravity S_i of the i^{th} body in respect to its own body fixed frame. Each body is associated with its mass m_i and inertia tensor $\bar{\boldsymbol{I}}_i$. The calculation of $\boldsymbol{M}$ demands for PKM a considerable computational effort. Even if elegant methodologies were proposed for deriving inertia matrices (Lin and Fang, 1999; Dulęba, 2002), these approaches remain interesting in a theoretical point of view. They seem to be developed for serial mechanisms and require a huge symbolic computation for PKM. They are not feasible for real-time applications. This explains, why practical application is till now associated with simple planar kinematics (Hesselbach *et al.*, 2004; Puchtler *et al.*, 2004) or with considerable simplification of the dynamics model (Courdourey, 1996; Honegger *et al.*, 1997).

In order to satisfy real-time requirements, a practical method is described in the following which provides efficient computation of the mass matrix. By formally splitting the manipulator into single kinematic chains with own minimal coordinates, the computational effort can be rigorously reduced, even for very complex mechanisms.

3. EFFICIENT COMPUTATION OF THE MASS MATRIX

The main idea of the approach is that the inertia matrix of the whole manipulator has to be a combination of the mass matrices of the incorporated kinematic substrings. It remains to prove and to quantify such relationship.

3.1 Definitions and Proofs

Considered is a general parallel manipulator as it is exemplarily given in Fig. 2. The vector of minimal coordinates $\boldsymbol{\lambda}$ is described with the displacement of the end-effector platform $\boldsymbol{r}_E = \left[\, x,\, y,\, z \,\right]$ and the vector of the orientation $\boldsymbol{o} = \left[\, \alpha,\, \beta,\, \gamma \,\right]$ as $\boldsymbol{\lambda} = \left[\, \boldsymbol{r}_E,\, \boldsymbol{o} \,\right]^\mathrm{T}$. The vector of the generalised velocities is defined as $\dot{\boldsymbol{\theta}} = \left[\, \boldsymbol{v}_E^\mathrm{T},\, \boldsymbol{\omega}_E^\mathrm{T} \,\right]^\mathrm{T}$ and includes the translational and angular velocities with reference to the cartesian frame. The minimal velocities are then different from the time derivative of $\boldsymbol{\lambda}$ ($\dot{\boldsymbol{\theta}} \neq \dot{\boldsymbol{\lambda}}$) (Merlet, 2000).

Formally, parallel manipulators can be considered as set of N_C coupled single kinematic chains (including the end-effector platform as single-body chain, see Fig. 3). Each chain j may contain N_j bodies and is considered in its own frame

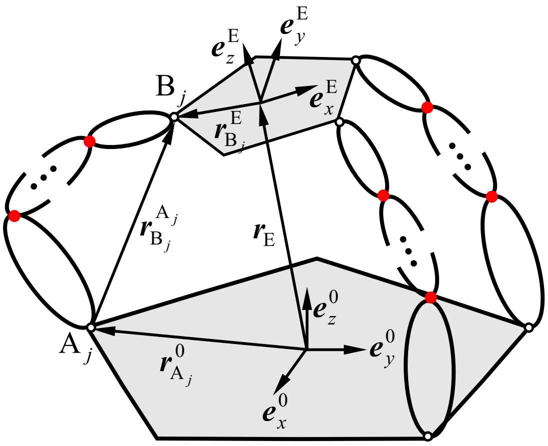

Fig. 2. Sketch of a parallel manipulator

$\{\boldsymbol{e}_x^j, \boldsymbol{e}_y^j, \boldsymbol{e}_z^j\}$ fixed in the base A_j or in the case of the end-effector in the TCP (Tool-Center-Point). The mass matrix can be rewritten to:

$$\boldsymbol{M} = \sum_{j=1}^{N_\mathrm{C}} \sum_{k=1}^{N_j} \left[m_k \left(\frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{\theta}}} \right)^\mathrm{T} \frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{\theta}}} + \left(\frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{\theta}}} \right)^\mathrm{T} \bar{\boldsymbol{I}}_k \frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{\theta}}} \right]. \tag{2}$$

To facilitate the handling of the equation, minimal coordinates $\boldsymbol{x}_j$ and minimal velocities $\dot{\boldsymbol{x}}_j$ for each kinematic chain j are introduced. For the serial parts, $\boldsymbol{x}_j$ and $\dot{\boldsymbol{x}}_j$ are the 3-dimensional displacements and velocities of the chain's ends B_j in respect to the chain's own frame(!). For the EE-Platform however minimal coordinates and velocities are identical with those of the manipulator: $\boldsymbol{x}_\mathrm{E} = \boldsymbol{\lambda}$ and $\dot{\boldsymbol{x}}_\mathrm{E} = \dot{\boldsymbol{\theta}}$.

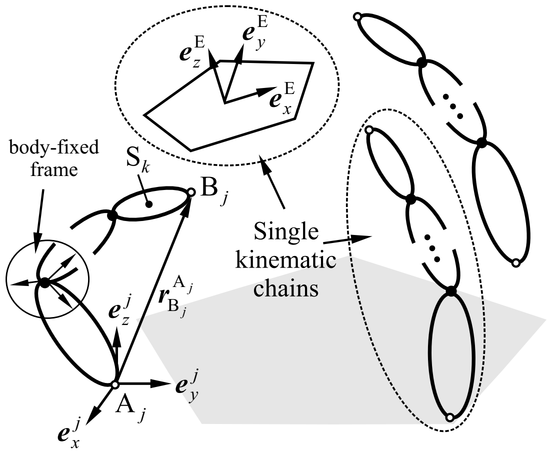

Fig. 3. Single kinematic chains of a parallel manipulator

Introducing the chain's minimal velocities into eq. (2) yields:

$$\boldsymbol{M} = \sum_{j=1}^{N_\mathrm{C}} \sum_{k=1}^{N_j} \left[m_k \left(\frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{x}}_j} \frac{\partial \dot{\boldsymbol{x}}_j}{\partial \dot{\boldsymbol{\theta}}} \right)^\mathrm{T} \frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{x}}_j} \frac{\partial \dot{\boldsymbol{x}}_j}{\partial \dot{\boldsymbol{\theta}}} + \left(\frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{x}}_j} \frac{\partial \dot{\boldsymbol{x}}_j}{\partial \dot{\boldsymbol{\theta}}} \right)^\mathrm{T} \bar{\boldsymbol{I}}_k \frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{x}}_j} \frac{\partial \dot{\boldsymbol{x}}_j}{\partial \dot{\boldsymbol{\theta}}} \right]. \tag{3}$$

By factorizing with the jacobians $\frac{\partial \dot{\boldsymbol{x}}_j}{\partial \dot{\boldsymbol{\theta}}}$, it results:

$$\boldsymbol{M} = \sum_{j=1}^{N_\mathrm{C}} \left[\left(\frac{\partial \dot{\boldsymbol{x}}_j}{\partial \dot{\boldsymbol{\theta}}} \right)^\mathrm{T} \sum_{k=1}^{N_j} \left[m_k \left(\frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{x}}_j} \right)^\mathrm{T} \frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{x}}_j} + \left(\frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{x}}_j} \right)^\mathrm{T} \bar{\boldsymbol{I}}_k \frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{x}}_j} \right] \frac{\partial \dot{\boldsymbol{x}}_j}{\partial \dot{\boldsymbol{\theta}}} \right]. \tag{4}$$

It's now recognizable that the second summation term is according to eq. (1) nothing but the mass matrix $\boldsymbol{M}_j$ of the j^{th} kinematic chain:

$$\boldsymbol{M}_j = \sum_{k=1}^{N_j} \left[m_k \left(\frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{x}}_j} \right)^\mathrm{T} \frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{x}}_j} + \left(\frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{x}}_j} \right)^\mathrm{T} \bar{\boldsymbol{I}}_k \frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{x}}_j} \right], \tag{5}$$

which is very much easier and efficient to compute (see also section 3.2). Besides, it is advantageous to consider the minimal velocities of the kinematic chains in respect to the cartesian frame $\dot{\boldsymbol{r}}_j = \left(\boldsymbol{R}_0^j\right)^{-1} \dot{\boldsymbol{x}}_j$, where $\boldsymbol{R}_0^j$ is the transformation matrix from the base frame $\{\boldsymbol{e}_x^0, \boldsymbol{e}_y^0, \boldsymbol{e}_z^0\}$ to the chain's local frame $\{\boldsymbol{e}_x^j, \boldsymbol{e}_y^j, \boldsymbol{e}_z^j\}$. Since $\boldsymbol{R}_0^j$ depends only from the geometric design of the mechanism, it results:

$$\frac{\partial \dot{\boldsymbol{x}}_j}{\partial \dot{\boldsymbol{\theta}}} = \boldsymbol{R}_0^j \frac{\partial \dot{\boldsymbol{r}}_j}{\partial \dot{\boldsymbol{\theta}}} = \boldsymbol{R}_0^j \boldsymbol{J}_{\mathrm{C},j}. \tag{6}$$

Inserting eq. (5) and (6) in eq. (4) yields the following compact form:

$$\boldsymbol{M} = \sum_{j=1}^{N_\mathrm{C}} \left[\boldsymbol{J}_{\mathrm{C},j}^\mathrm{T} \boldsymbol{R}_j^0 \boldsymbol{M}_j \boldsymbol{R}_0^j \boldsymbol{J}_{\mathrm{C},j} \right]. \tag{7}$$

The calculation of the Inertia matrix of the robotic manipulator is reduced to the following simple steps:

- the inverse kinematics of the manipulator yields minimal coordinates and velocities for single chains.
- with the inverse kinematics of every chain, the velocities of the bodies are obtained.
- calculation of $\boldsymbol{M}_j$ according to eq. (5).
- transformation and summation of the obtained inertia matrices $\boldsymbol{M}_j$ according to eq. (7).

In the following subsection the necessary kinematic operations are explained.

3.2 Kinematic Analysis

Starting from the minimal coordinates $\boldsymbol{\lambda}$ and velocities $\dot{\boldsymbol{\theta}}$ of the manipulator, the inverse kinematics calculates minimal coordinates and velocities of single chains by

$$\boldsymbol{r}_j = -\boldsymbol{r}^0_{\mathrm{A}_j} + \boldsymbol{r}_{\mathrm{E}} + \boldsymbol{R}^0_{\mathrm{E}}\boldsymbol{r}^{\mathrm{E}}_{\mathrm{B}_j}, \tag{8}$$

$$\dot{\boldsymbol{r}}_j = \boldsymbol{v}_{\mathrm{E}} + \boldsymbol{\omega}_{\mathrm{E}} \times \boldsymbol{r}^{\mathrm{E}}_{\mathrm{B}_j}, \tag{9}$$

where $\boldsymbol{R}^0_{\mathrm{E}}$ is the orientation matrix of the end effector (Merlet, 2000). Inserting eq. (9) in eq. (6) yields

$$\boldsymbol{J}_{\mathrm{C},j} = \left[\, \boldsymbol{I} \;\; -\tilde{\boldsymbol{r}}^{\mathrm{E}}_{\mathrm{B}_j} \,\right]. \tag{10}$$

By formally splitting the manipulator into basic chains with reduced number of minimal coordinates, the computation effort is considerably reduced. This is advantageous for the calculation of the translational and rotational jacobians of each Body according to

$$\boldsymbol{J}^{jk}_{\mathrm{T}} = \frac{\partial \bar{\boldsymbol{v}}_k}{\partial \dot{\boldsymbol{x}}_j} \text{ and } \boldsymbol{J}^{jk}_{\mathrm{R}} = \frac{\partial \bar{\boldsymbol{\omega}}_k}{\partial \dot{\boldsymbol{x}}_j}. \tag{11}$$

The velocities of the bodies are obtained by solving the inverse kinematic of each chain. The body-fixed coordinate frames (see also Fig. 3) can be defined accordingly to the modified Denavit-Hartenberg (MDH)-Notation (see (Grotjahn *et al.*, 2004) for further details). This allows a recursive calculation of the quantities of motion starting from eq. (8) and eq. (9):

$$\boldsymbol{\omega}_k = \boldsymbol{\omega}_{k-1} + \sigma_k \dot{\theta}_k, \tag{12}$$

$$\boldsymbol{v}_k = \boldsymbol{v}_{k-1} + \boldsymbol{\omega}_{k-1} \times \boldsymbol{r}^{k-1}_k + \bar{\sigma}_k \dot{d}_k \boldsymbol{z}, \tag{13}$$

where $\bar{\sigma}_k = 1 - \sigma_k$ ($\sigma_k = 1$ for revolute, and $\sigma_k = 0$ for prismatic joint). The velocities of the body's center of gravity are then simply deduced by:

$$\bar{\boldsymbol{\omega}}_k = \boldsymbol{\omega}_k \text{ and } \bar{\boldsymbol{v}}_k = \boldsymbol{v}_k + \boldsymbol{\omega}_{k-1} \times \boldsymbol{r}^k_{S_k} \tag{14}$$

where $\boldsymbol{r}^k_{S_k}$ is the location vector of the body's center of gravity. It is important to notice, that eq. (12, 13, 14) are not involved in the proper calculation of the mass matrix. They are necessary in a preliminary step of kinematic analysis to determine the analytical form of each body's jacobians $\boldsymbol{J}^{jk}_{\mathrm{T}}$ and $\boldsymbol{J}^{jk}_{\mathrm{R}}$.

4. EVALUATION

The methodology for calculating the inertia matrix and its integration in the control was implemented on the innovative hexapod PaLiDA (Parallelkinematik with Linear Direct Drives). The machine was constructed at the Institute of Production Engineering and Machine Tools of the University of Hannover. Our research aims qualifying PKM for machining and manipulation tasks with high dynamic requirements and low process forces. For that purpose, PaLiDA is equipped with electrical linear direct drives. They have the advantages of reduced mechanical components, no backlash and low inertia with a minimized number of wear parts. Furthermore, higher control bandwidth and extremely high accelerations can be achieved (Grotjahn *et al.*, 2004; Denkena *et al.*, 2004).

Fig. 4. PKM PaLiDA - a: Test bed at the Hannover Industrial Fair 2001, b: CAD-model, c: single strut with integrated linear direct drive

PaLiDA is modelled with 19 bodies. In addition to the movable platform, each of the six struts is composed of three bodies: a movable Cardan ring, a stator and a slider(see Fig. 4). To integrate the inertia-matrix into the control scheme, the methodology described in the former section was implemented for real-time calculation within a rate of 0,5 ms. The computational effort in a sense of executed operations is listed for single steps in Table 1. The presented approach is compared with two others effective methods known from literature.

Table 1. Computational effort for the calculation of the mass matrix

	$+/-$	$\times/\div$
$1\times$ inverse kinematics: eq. (8, 9)	120	156
$6\times$ mass matrix of a strut: eq. (5)	29	54
$6\times$ transformations: eq. (7)	23	44
$1\times$ mass matrix of EE: eq. (5)	15	17
summation : eq. (7)	252	-
total with proposed approach	699	761
approach from (Dulęba, 2002)	2365	2964
approach from (Lin and Fang, 1999)	2550	3839

The proposed approach is significantly more efficient than those provided in (Lin and Fang, 1999) and in (Dulęba, 2002). The computational cost is considerably reduced and a real time application is possible even with very small sample rates. Although the methods from (Lin and Fang, 1999) and (Dulęba, 2002) are very interesting, their efficiency could not be proved for parallel manipulators.

The experimental application is now demonstrated on a benchmark motion with high dynamics. The path consists in a full circle (diameter = 0,3 m) in the middle of the workspace and is inclined by 45° in respect to the inertial x-axis. The horizontal end-effector is accelerated from the standstill, to reach the velocity of 1 ms^{-1}. The evaluation of the mass matrix along that path is depicted in Fig. 5, where the effective inertia of three arbitrarily chosen actuators are shown. It is

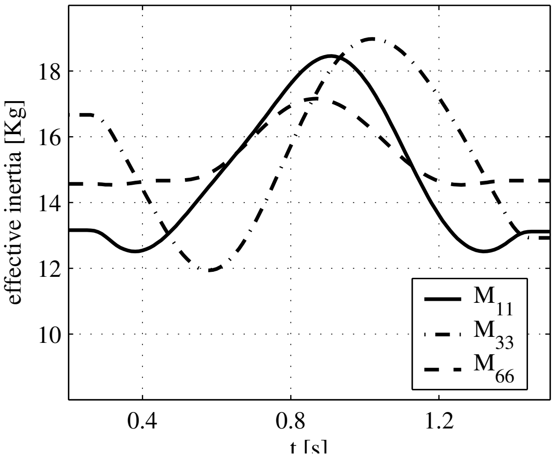

Fig. 5. Variations of the inertia while a circular motion

easy to notice, that the effective inertia of some actuators varies in a range of approximately 60 %. It is obvious, that a static parameterized model for actuator inertia will lead to significant and pose dependent errors. For the same benchmark motion, the results of different control strategies integrating the coupled and nonlinear dynamics are discussed with experimental results in the following section.

5. EXPERIMENTAL RESULTS

In the following three control concepts are compared exerimentaly: conventional single-joint control, computed-force (or feedforward) control and computed-force control with additional axes decoupling by using the mass matrix. The performance of the implemented axes decoupling control (see Fig. 1) is demonstrated considering the circular motion as an example. In Fig. 6 control errors $\Delta q_{a,i}$ of 3 arbitrarily chosen actuators are depicted. Conventional single-joint PID-control is clearly improper and unsuitable for operating PKM at high dynamics. This is obvious especially during acceleration phases, where simple PID control produces intolerable performance. Also during periods with constant velocity, single-joint-control is unsatisfactory. The important tracking errors are due to the very high dynamics, while circular motion with a TCP-velocity of 1 ms^{-1}. Notice that the velocity range tested in (Honegger *et al.*, 2000) was limited to 0.4 ms^{-1}.

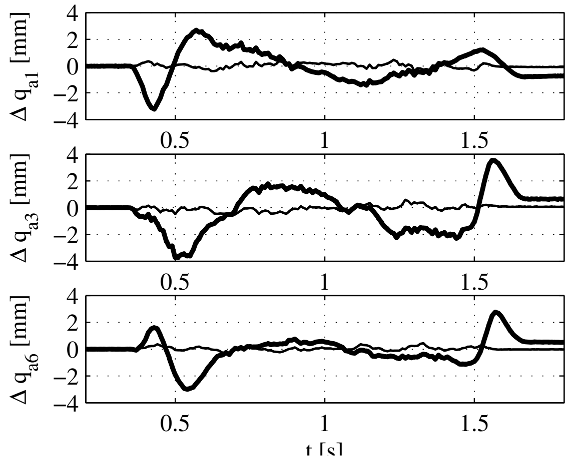

Fig. 6. Control errors in single actuators. Comparison of conventional single-joint PID-control (thick line) with feedforward decoupling control (thin Line)

The importance of considering the pose dependent mass matrix is demonstrated in Fig. 7 (Notice the different axis scaling to Fig. 6!). For the same motion, tracking errors were investigated with a control using only feedforward and a control with additional axes decoupling by regarding varying inertia (see Fig. 1). The integration of the computation of the mass matrix into the control concept is definitely more advantageous. Besides, the depicted results are consistent with the calculated effective inertia shown in Fig. 5. The more the effective inertia of the actuator varies, the more important become the related tracking errors if no decoupling control is used. This can be observed for the actuators nr. 1 and nr. 3. For actuator nr. 6, the performances of both control concepts are more close, because the corresponding effective inertia is less varying during the investigated motion.

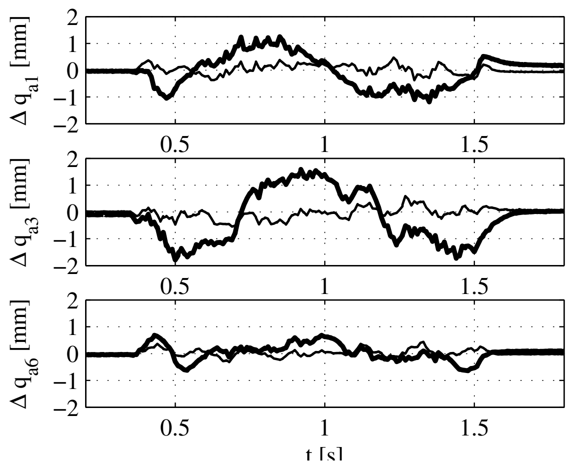

Fig. 7. Control errors in single actuators. Comparison of feedforward control without axes-decoupling (thick line) with feedforward decoupling control (thin Line).

6. CONCLUSIONS

The aim of this paper was to prove the necessity of considering nonlinear and coupled dynamics for accurate control of parallel manipulators. Decoupling control is performed by the integration and real-time calculation of the mass matrix. A high computational efficient approach was therefore presented, which allows such task without any model simplifications. The experimental evaluation was successfully performed on the complex 6-DOF-manipulator PaLiDA within very small sample rates. Experimental examples of high dynamic motions have been used to demonstrate the crucial role of decoupling control in increasing the tracking accuracy. Thereby, three different control concepts were evaluated. The simultaneous consideration of the nonlinear dynamics by feedforward control and the decoupling of the axes by computing the mass matrix yields best results.

REFERENCES

Abdellatif, H., M. Grotjahn and B. Heimann (2004). Path accuracy improvement of parallel kinematic structures by the identification of friction and rigid-body dynamics. In: *Proc. of the 4th Chemnitz Parallel Kinematics Seminar PKS 2004*. Chemnitz, Germany. pp. 217–236.

Chae, A. H., C.G. Atkeson and J.M. Hollerbach (1986). Experimental determination of the effect of feedforward control on trajectory tracking errors. In: *Proc. of the 1986 IEEE Int. Conf. on Robotics and Automation*. pp. 55–60.

Courdourey, A. (1996). Dynamic modelling and mass matrix evaluation of the delta parallel robot for axes decoupling control. In: *Proc. of the 1996 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS 96*. Osaka, Japan. pp. 1211–1218.

Denkena, B., H. Grendel and C. Holz (2004). Model based feedforward and state control of the parallel kinematics palida. In: *Proc. of the 4th Chemnitz Parallel Kinematics Seminar PKS 2004*. Chemnitz, Germany. pp. 185–202.

Dulęba, I. (2002). Structural properties of inertia matrix and gravity vector of dynamics of rigid manipulator. *Journal of Robotic Systems* **19**(11), 555–567.

Grotjahn, M., B. Heimann, J.B. Kuehn and H. Grendel (2004). Dynamics of robot with parallel kinematic structures. In: *Proc. of the 11th IFToMM World Congress in Mechanism and Machine Science*. Tjanjin, China. pp. 1689–1693.

Hesselbach, J., J. Pietsch, C.C. Bier and O.T. Becker (2004). Model-based control of plane parallel robots - how to choose the appropriate approach?. In: *Proc. of the 4th Chemnitz Parallel Kinematics Seminar PKS 2004*. Chemnitz, Germany. pp. 211–232.

Honegger, M., A. Cordourey and E. Burdet (1997). Adaptive control of the hexaglide, a 6 dof parallel manipulator. In: *Proc. of the 1997 IEEE Int. Conf. on Robotics and Automation*. Albuquerque. pp. 543–548.

Honegger, M., R. Brega and G. Schweitzer (2000). Application of a nonlinear adaptive controller to a 6 dof parallel manipulator. In: *Proc. of the 2000 IEEE Int. Conf. on Robotics and Automation*. San Francisco. pp. 1930–1935.

Khalil, W. and E. Dombre (2002). *Modeling, Identification and Control of Robots*. Hermes Penton Science. London.

Lin, S.-K. and C.-J. Fang (1999). Efficient formulations of the manipulator inertia matrix in terms of minimal linear combinations of inertia parameters. *Journal of Robotic Systems* **16**(12), 679–695.

Merlet, J.P. (2000). *Parallel Robots, Solid Mechanics and its Applications*. Kluwer Academic Publishers. Doderecht, Netherlands.

Puchtler, T., C. Hamm and K. Geissdoerfer (2004). Pkm productivity optimized by conserning mass inertia during motion planning. In: *Proc. of the 4th Chemnitz Parallel Kinematics Seminar PKS 2004*. Chemnitz, Germany. pp. 203–210.