

REALTIME COORDINATED REDUNDANT MOTION OF A NONHOLONOMIC MOBILE MANIPULATOR

Günter Schreiber and Gerd Hirzinger

DLR Institute of Robotics and Mechatronics

82230 Wessling, Germany

Guenther.Schreiber@dlr.de

Abstract In this paper, the problem of coordinated motion control of a non-holonomic mobile platform with a serial manipulator mounted on top of the platform is addressed. The redundancy resolution algorithm bases on a constraint optimization approach, where the non-holonomic behaviour of the platform is modeled into inequality constraints leading to a convex optimization problem.

Keywords: Non-holonomy, Redundancy, Constraint Optimization, Realtime

1. Introduction

In service robotic applications the combination of a mobile platform with a serial manipulator is of big interest. Especially the coordinated motion of the two subsystems will be addressed within this paper, whereas the mobile platform has a non-holonomic behaviour. The most known applications about coordinated motions use a holonomic platform (Khatib et al., 1996) (Yamamoto & Yun, 1994) (Watanabe, Sato, Izumi, & Kunitake, 2000), which then allows the application of redundancy resolution algorithms based on the well known Moore-Penrose pseudo inverse (D.E. Whitney, 1969). Unfortunately in the case of non-holonomic behaviour, those redundancy resolution algorithms cannot be applied easily.

In our approach (section 3), the non-holonomic couplings of the joints are formulated into a convex optimization problem with linear inequality constraints, so that coordinated motion can be computed in realtime. This kind of algorithms base on (sequential) quadratic programming and constraint optimization problems, which have been studied by several authors (Schreiber, Otter, & Hirzinger, 1999) (Schlemmer, 1997) (Cheng, Chen, & Sun, 1994).

The paper is organized as follows: In the next section, the utilized platform is sketched. Then the redundancy resolution algorithm based

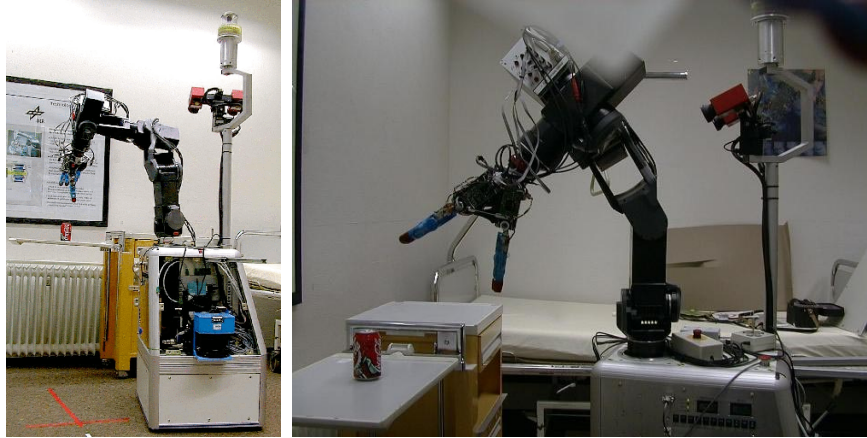


Figure 1. The DLR Lightweight Robot mounted on a mobile platform

on constraint optimization is outlined. Thereafter the linear inequality constraints are developed, and numerical results are presented.

2. Description and modelling of the mobile platform

The utilized mobile platform is “omni directional”, since it is able to move in any direction, but not at any time (Saldic, Hanebeck, & Schmidt, 2000). The entire system is presented in figure 1. The platform itself is assembled with four wheelsets, which are located at the four corners of the platform. Those wheelsets are each controllable in two Degrees of Freedom (DoF). The fact, that the wheels have no offset between their steering axis and the point, where they touch the ground, is the reason for the non-holonomy of this setting. For our modelling purposes, we assume that the eight DoF are subject to low level control of the platform, and reduce the problem to three position coordinates $\mathbf{q}_p = (\xi, \zeta, \theta)^T$ which is shown in figure 2(a). Furtheron, as the entire manipulator system is considered, the joint vector $\mathbf{q} = (\mathbf{q}_p^T \mathbf{q}_m^T)^T$ is defined, which is constructed from the joint vector of the mobile platform \mathbf{q}_p and the joint vector of the serial manipulator \mathbf{q}_m .

We assume, that the low level control of the platform chooses a configuration of the wheels so that motion will be possible. And furthermore that it computes a normalized direction vector \mathbf{d} within the space of $\dot{\mathbf{q}}_p$ in which the wheels are facing. As the mobile platform is in motion, this will be nothing else than the normalized current speed in joint space of the platform.

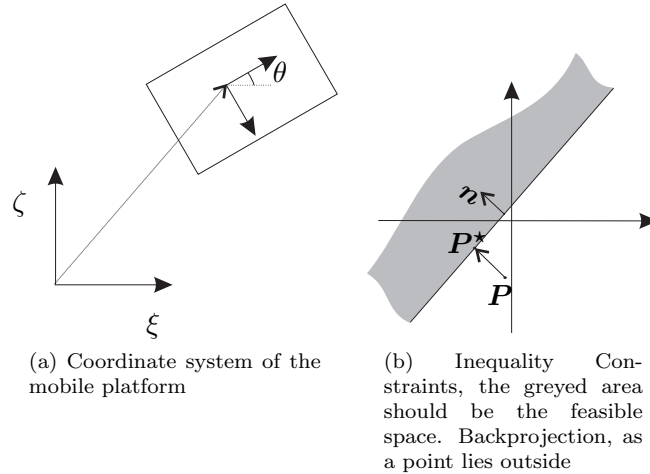


Figure 2. Coordinate system of the platform and Visualization of linear inequality constraints

3. Constraint optimization for redundancy resolution

In the sequel the problem of redundant inverse kinematics is reviewed and our current system based on constraint optimization is outlined. Various different authors dealt with the solution of the redundant inverse kinematic problem (see (Siciliano, 1990) for an overview). Cartesian position and orientation \mathbf{x} of the end-effector can be described as a function of the vector of joint variables \mathbf{q} of the manipulator.

$$\mathbf{x} = \mathbf{f}(\mathbf{q}) \quad (1)$$

While equation (1) can be obtained easily, the inverse problem is crucial. In the redundant case it is generally not possible to find an inverse mapping \mathbf{f}^{-1} . Instead of constructing an inverse function $\mathbf{g}(\mathbf{x})$ with $\mathbf{f}(\mathbf{g}(\mathbf{x})) = \mathbf{x}$ analytically, the problem is often reformulated in the velocities utilizing the partial derivative of $\mathbf{f}(\mathbf{q})$, which leads to the wellknown Jacobian matrix $\mathbf{J} = \frac{\partial \mathbf{f}(\mathbf{q})}{\partial \mathbf{q}}$.

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \quad (2)$$

In our current system, a linear least square based constraint optimization is utilized (Schreiber et al., 1999; Cheng et al., 1994; Park, Chang, & Salisbury, 1997). The major advantage by utilizing the constraint optimization approach is that physical constraints, like joint speed limits or non-holonomic constraints, can be enforced in a natural and simple

way via inequality constraints. Using the incremental formulation

$$\dot{\mathbf{q}} \approx \frac{\Delta \mathbf{q}}{T_a} \quad (3)$$

linear conditions of the form $\mathbf{A}\Delta \mathbf{q} = \mathbf{a}$ can be chosen for the nullspace motion. So the inverse kinematic problem can be formulated as an optimization problem $\min_{\Delta \mathbf{q}} L_{IK}(\Delta \mathbf{q})$ for the cost function

$$L_{IK}(\Delta \mathbf{q}) = \|\mathbf{A}\Delta \mathbf{q} - \mathbf{a}\|_2 \quad (4)$$

with the equality constraint as main task

$$\Delta \mathbf{x} = \mathbf{J}\Delta \mathbf{q} = \mathbf{x}_d - \mathbf{x} \quad (5)$$

Herein the symbol $\|\cdot\|_2$ denotes the quadratic norm, \mathbf{x}_d the desired cartesian, T_a is the sample time and the symbol Δ in front of a variables denotes an increment. Notice that the main task forms in fact the constraint of a linear optimization problem here. Additional constraints in form of inequalities can be added. Those are defined by the following wellknown expression of (hyper-) planes (see figure 2(b)):

$$\mathbf{n}^T \Delta \mathbf{q} \geq D \quad (6)$$

whereas \mathbf{n} is the normal vector defining the plane and D is the shortest distance of the plane to the origin.

Also we define the rule, if point \mathbf{P} is outside the allowed area, that means expression (6) does not hold, a new point \mathbf{P}^* will be defined such that \mathbf{P} is pushed back on the shortest path into the allowed area:

$$\mathbf{P}^* = \mathbf{P} + (\mathbf{n}^T \mathbf{P} - D)\mathbf{n} \quad (7)$$

This will be used to constrain the solutions for particular joints e.g. due to hardware restrictions as maximum joint speeds or joint limits.

$$\Delta \mathbf{q}_{min} \leq \Delta \mathbf{q} \leq \Delta \mathbf{q}_{max} \quad (8)$$

Any particular cost function $h(\mathbf{q})$ can be minimized locally by utilizing the $n \times n$ -identity matrix $\mathbf{E}_{n \times n}$ and the gradient of the cost function, whereas n is the number of joints.

$$\mathbf{A} = \mathbf{E}_{n \times n}, \quad \mathbf{a} = -\frac{\partial h(\mathbf{q})}{\partial \mathbf{q}} \quad (9)$$

It is now possible to combine multiple behaviors for the null-space motion by choosing

$$\mathbf{A} = \begin{pmatrix} w_1 \mathbf{A}_1 \\ \vdots \\ w_l \mathbf{A}_l \end{pmatrix}, \quad \mathbf{a} = \begin{pmatrix} w_1 \mathbf{a}_1 \\ \vdots \\ w_l \mathbf{a}_l \end{pmatrix} \quad (10)$$

The particular equations $\mathbf{A}_i \Delta \mathbf{q} = \mathbf{a}_i$ are weighted by the factors w_i , which define how much an error in the i^{th} equation affects the value of the cost function $L_{IK}(\Delta \mathbf{q})$. One very interesting feature of this formulation is that the main task can also be included into the cost function instead of constraining the optimization if the problem gets ill-posed, e.g. due to singular configurations (Schreiber et al., 1999). Other cost functions can be found in (Schreiber, Ott, & Hirzinger, 2001) and cost functions concerning interactive behaviour in (Schreiber & Hirzinger, 2000). Standard numerical algorithms exist to solve this special convex optimization problem in a finite number of steps in a reliable way, see e.g. (Lawson & Hanson, 1974; Björk, 1996) for details.

3.1 Metrics in coordinates

It should be noted, that in this system two different units (rotational and translational) in joint space are compared to each other, which is not obviously valid. In fact the increments $\Delta \mathbf{q}$ are minimized with a quadratic norm. This implies that there is an implicit metric, with respect to joint space. This metric can be influenced by a non-singular weighting matrix \mathbf{W} , so that $\|\mathbf{W} \Delta \mathbf{q}\|_2$ is minimized. Therefore $\Delta \mathbf{q}_W$ is defined

$$\Delta \mathbf{q}_W = \mathbf{W} \Delta \mathbf{q} \quad (11)$$

$$\Delta \mathbf{q} = \mathbf{W}^{-1} \Delta \mathbf{q}_W \quad (12)$$

and used instead of $\Delta \mathbf{q}$ for the solution of the constraint optimization problem. This kind of metrics are considered to be applied in sequel.

4. Inequality Constraints for the non-holonomy

To embed the non-holonomic constraints into the previously introduced optimization algorithm, some linear inequality constraints have to be constructed according to (6). In principle the configuration of the mobile platform defines a one DoF manifold for an instantaneous motion within the three dimensional joint space \mathbf{q}_P . But under consideration of increments during the sample time T_a , the configuration can be changed, so that the feasible motion varies around the current direction \mathbf{d} of the mobile platform (see figure 3(a), for the sake of simplicity, only two of the three dimensions will be outlined in the figures). This behaviour will be exploited to construct linear inequality constraints for generating coordinated motion in realtime. Due to linearization and convexity effects, the allowed area will be smaller than the feasible area, which the platform could reach during a time interval. But for (mechanical) stability reasons the maximum speeds and accelerations are reduced anyway.

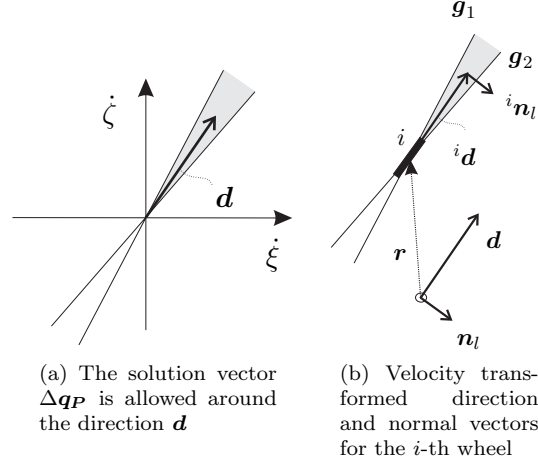


Figure 3. Inequality Constraints

The space of the feasible solution $\Delta \mathbf{q}_{P,i}$ to the optimization problem can be thought of a pyramidal or conical shape around the current direction of the platform. The opening angle of that shape is restricted by the steering angular velocity of the wheels. So, this shape is approximated by a number of planes within $\Delta \mathbf{q}_P$.

4.1 Inequality constraints when platform is in motion

As the mobile platform is moving, the new computed increment $\Delta \mathbf{q}_{P,i}$ is dependent on the direction \mathbf{d} . To construct a vector normal to \mathbf{d} , we will use the vector product, since $\Delta \mathbf{q}_P$ is of dimension three. We will use the unit vectors $\mathbf{u}_1 = (1 \ 0 \ 0)^T$, $\mathbf{u}_2 = (0 \ 1 \ 0)^T$ and $\mathbf{u}_3 = (0 \ 0 \ 1)^T$ as generators, whereas we select that unit vector \mathbf{u}_j^T , which has the smallest scalar product to \mathbf{d} , in order to guarantee that the generator is not parallel to \mathbf{d} :

$$\mathbf{n}_P = \mathbf{u}_j \times \mathbf{d} \quad (13)$$

To construct k pyramidally shaped inequality constraints, the vector \mathbf{n}_P is rotated around the direction vector \mathbf{d} and then rotated around vector \mathbf{u}_j with the angle α , which is dependent on the maximum steering angular velocity twice for each of the sides. D of (6) is set to zero:

$$\mathbf{n}_l = \left(\text{rot}_{\mathbf{d}} \left(\frac{l}{k} 360^\circ \right) \mathbf{n}_P \right) \quad (14)$$

$$\mathbf{n}_{l,\alpha} = \left(\text{rot}_{\text{rot}_{\mathbf{d}}(\frac{l}{k}360)} \mathbf{u}_j(\alpha_l) \mathbf{n}_l \right) \quad (15)$$

$$\mathbf{n}_{l,\alpha}^T \Delta \mathbf{q}_P \geq 0 \quad (16)$$

whereas for the counting index l holds $0 \leq l < k$. In our case, it is sufficient utilizing only $k = 4$ planes of the k -sided pyramid. $\text{rot}_{\mathbf{u}_j}(\alpha)$ is a rotation around the axis of vector \mathbf{u}_j with the angle α .

4.2 Computation of angle α_l

The angle α_l can be estimated from the current steering angle of each wheel. Therefore, the current direction \mathbf{d} and the constructing normal vector \mathbf{n}_l is velocity transformed into the i -th wheels coordinate system,

$$\begin{pmatrix} {}^i\dot{\xi} \\ {}^i\dot{\zeta} \end{pmatrix} = \begin{pmatrix} \dot{\xi} \\ \dot{\zeta} \end{pmatrix} + \dot{\theta} \begin{pmatrix} -r_2 \\ r_1 \end{pmatrix} \quad (17)$$

leading to ${}^i\mathbf{d}$ and ${}^i\mathbf{n}_l$ (see fig. 3(b)), whereas \mathbf{r} is the vector connecting the platform coordinate system to the i -th wheel. Note that, due to the assumptions concerning the platform controller, in the resulting vectors only translational components are considered. For the i -th wheel, depending on its state, two lines $\mathbf{g}_{1,2}$ depending on its steering angle velocity can be computed, denoting the maximum variation of the steering angle within the time interval. Now a parameter $\lambda_i > 0$ is computed, which depicts the point, where the line $\mathbf{g}_{1,2}$ is intersected by ${}^i\mathbf{d} + \lambda_i {}^i\mathbf{n}_l$. From all wheels, the smallest $\lambda_i > 0$ is chosen to compute the angle $\alpha_l = \text{atan}(\lambda_i)$.

4.3 Inequality constraints as platform has low speed

Special attention has to be paid, if the platform has with low speed or is at rest. In this case, the platform could physically revert it's direction. In this case at least two iterations will be needed (for details about iterative formulation see (Schreiber et al., 1999)) per sample time. A prismatic shaped constraint is used in the first iteration (see fig. 4). Therefore the opening angle α_l is set to zero, and the distance parameter D of equation (6) is set to a small value $\varepsilon_l > 0$, which correlates to the opening of the corresponding pyramid.

$$\mathbf{n}_l = \left(\text{rot}_{\mathbf{d}}(\frac{l}{k}360^\circ) \mathbf{n}_P \right) \quad (18)$$

$$\mathbf{n}_l^T \Delta \mathbf{q}_P \geq -\varepsilon_l \quad (19)$$

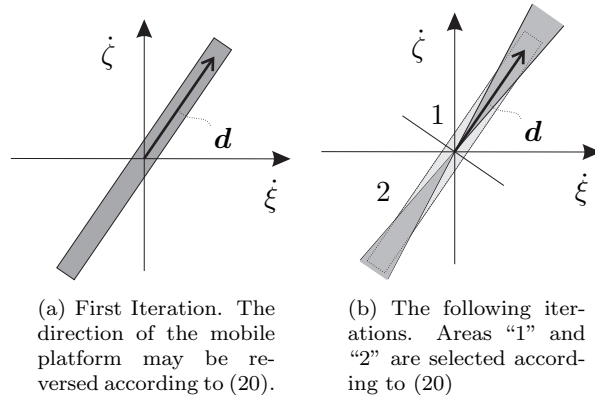


Figure 4. Inequality Constraints for low speed case

whereas $0 \leq l < k$. For the next iteration, the proper branch is selected according to the iterative solution $\Delta \mathbf{q}_{P_i}^0$. This is performed by the dot product:

$$\mathbf{d}^T \Delta \mathbf{q}_{P_i} \geq 0 \quad (20)$$

If condition (20) holds, the inequality constraints are constructed as in the previous section (see fig. 4(b) - denoted as area 1). If condition (20) is false, the direction vector \mathbf{d} is reversed and the inequality constraints are constructed as in the previous section (see fig. 4(b) - denoted as area 2). The iterative solution vector $\Delta \mathbf{q}_i^0$, which is used as starting point for the next iteration is projected back into the feasible space utilizing equation (7).

5. Numerical results

The presented approach is implemented on our system. To demonstrate the non-holonomic effect best, the manipulator system has to follow a trajectory, for which the nonholonomic constraint is perpendicular to the requested motion. In figure 5, the manipulator is visualized during motion. In the upper left corner, the starting configuration is shown. The wheels are assumed to face along the greyed arrow, the trajectories of the endeffector and the position of the platform are displayed. If no non-holonomic constraint would be active, the mobile platform would follow a straight line, but with this constraint active, the mobile platform has to follow some curved trajectories due to the reconfiguration time of the wheels. This effect is obvious during the depicted motion. The motion falls into two phases. In the first phase, an endeffector motion is shown, in the second phase a nullmotion is performed to converge to a chosen configuration.

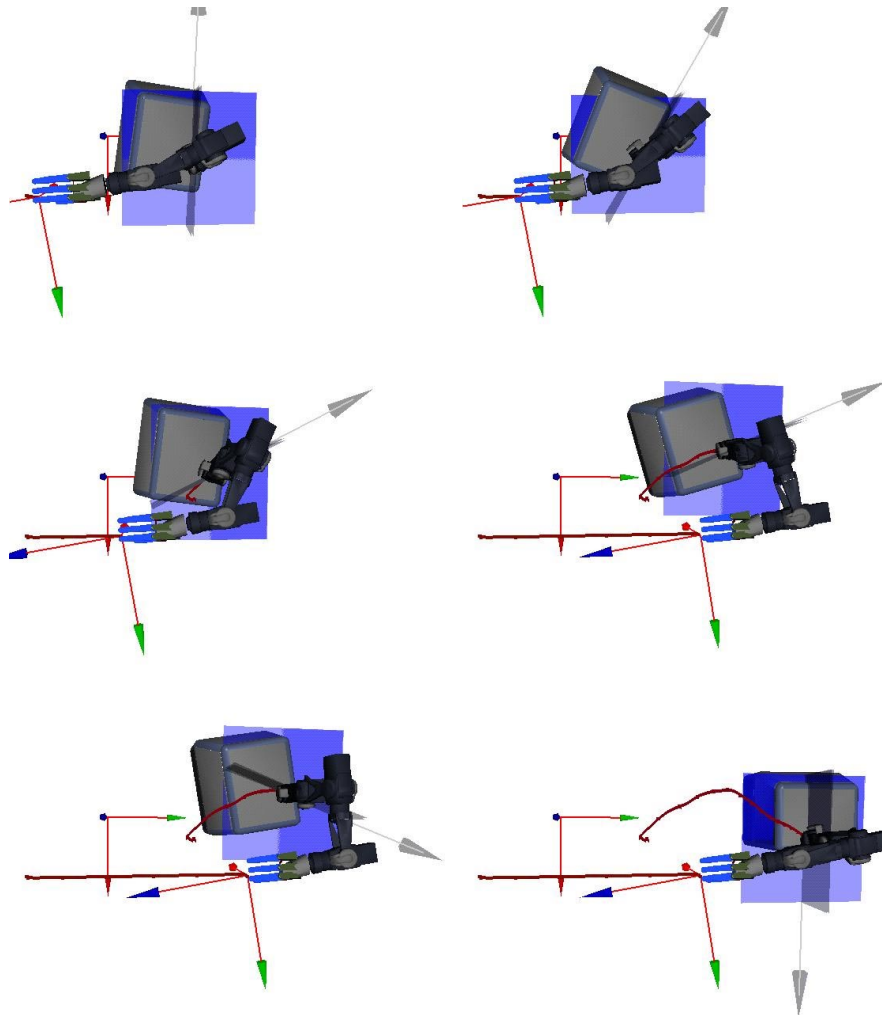


Figure 5. Snapshots of the sample trajectory. The manipulator's non-holonomic constraint faces along the greyed arrow. The linear inequality constraints are visualized as planes during the motion. One can see the curved motion of the first three axes as a consequence of the non-holonomic coupling of those joints. If no non-holonomic constraint were active, the trace of the platform would follow a straight line quite parallel to the trace of the endeffector. The first two rows of snapshots show an endeffector motion, the last row shows a subsequent nullmotion to a desired configuration.

6. Conclusion

Combined mobile manipulator systems, as the one presented, will be of great interest in service robotic applications. We have presented an

approach to embed non-holonomic constraints into redundancy resolution systems for realtime purposes. The non-holonomic constraints are therefore squeezed into a convex optimization problem. The mobile manipulator utilizing this approach will be presented on Hannover Fare Exhibition 2002.

Acknowledgement

This work was funded by “BMBF-Leitprojekt MORPHA” Number ITL 0902 E.

References

- Björk, A. (1996). *Numerical Methods for Least Squares Problems*. SIAM.
- Cheng, F.-T., Chen, T., & Sun, Y.-Y. (1994). Resolving manipulator redundancy under inequality constraints. *IEEE Tr.act. on Robotics and Automation*, 10(1), 65–71.
- D.E. Whitney. (1969). Resolved motion rate control of manipulators and human prostheses. *IEEE Tr.act. Man-Machine Systems*, 10, 47–53.
- Khatib, O., Yokou, K., Chang, K., Ruspini, D., Holmberg, R., & Casal, A. (1996). Coordination and decentralized cooperation of multiple mobile manipulator. *J. Robotic Systems*, 13(11), 755–764.
- Lawson, C., & Hanson, R. (1974). *Solving Least Squares Problems*. Prentice Hall.
- Park, K. C., Chang, P. H., & Salisbury, J. K. (1997). A unified approach for local resolution of kinematic redundancy with inequality constraints and its application to nuclear power plant. In *ICRA, April 1997* (pp. 766–773).
- Saldic, N., Hanebeck, U., & Schmidt, G. (2000). Systems for omnidirectional mobile robots. In *Proceedings 9 th International Workshop on Robotion in Alpe - Adria - Danube Region* (pp. 11–16).
- Schlemmer, M. (1997). *Eine SQP-Rückwärtskinematik zur interaktiven, semi-autonomen Bahnführung kinematisch redundanter Manipulatoren*. Unpublished doctoral dissertation, Gerhard-Mercator-Universität, Gesamthochschule Duisburg.
- Schreiber, G., & Hirzinger, G. (2000). An intuitive interface for nullspace teaching of redundant robots. In J. Lenarčič & M. Stanišić (Eds.), *Advances in Robot Kinematics* (pp. 209–216). Portoroz/ Slovenia: Kluwer Academic Publishers, Dordrecht.
- Schreiber, G., Ott, C., & Hirzinger, G. (2001). Interactive redundant robotics: Control of the inverted pendulum with nullspace motion. In *IROS 2001* (pp. 158–164).
- Schreiber, G., Otter, M., & Hirzinger, G. (1999). Solving the singularity problem of non-redundant manipulator by constraint optimization. In *IROS 1999, Kyongju, Korea* (pp. 1482–1488).
- Siciliano, B. (1990). Kinematic Control of Redundant Robot Manipulators: A Tutorial. *Journal of Intelligent and Robotic Systems*, 3, 201–212.
- Watanabe, K., Sato, K., Izumi, K., & Kunitake, Y. (2000). Analysis and control for an omnidirectional mobile manipulator. *J. of Int. and Rob. Systems*, 27, 3–20.
- Yamamoto, Y., & Yun, X. (1994). Coordinating locomotion and manipulation of manipulator. *IEEE Tr.act. on Automation and Control*, 39(6), 1326–1332.