Solving the Singularity Problem of non-redundant Manipulators by Constraint Optimization

Günter Schreiber, Martin Otter and Gerd Hirzinger

German Aerospace Center - DLR
Institute for Robotics and System Dynamics
82230 Wessling, Germany
E-mail: Guenter.Schreiber@dlr.de

Abstract — *A solution to the singularity problem of a non–redundant robot is proposed by reformulating the inverse kinematic problem as a constraint optimization problem. The main idea is to allow a cartesian error in a certain subspace in the vicinity of a singuarity and to minimize this error subject to operational constraints such as maximum motor speeds. As a result, in every sampling instant a series of linear least squares problems with linear equality and inequality constraints have to be solved. This task can be carried out on a Pentium processor within a few milliseconds. The new method is demonstrated at hand of some experiments with an industrial robot.*

1 Introduction

While traveling along a cartesian trajectory within the vicinity of a kinematic singularity, the manipulators Jacobian matrix is ill–conditioned and looses rank at the singular point, resulting in joint speeds and accelerations which exceed the physical limits of the corresponding manipulator. This problem can be overcome by (a) staying exactly on the desired trajectory and scaling the desired travelling time such that the robot moves slower [3] or by (b) explicitly deviating from the desired trajectory. Both approaches are meaningful for respective task problem classes.

Keeping exact cartesian position and orientation is useful for applications, where the path has been preplanned and to be kept explicitly, e.g. where collision safety has to be guaranteed. On the other hand, e.g. for applications of dispensing glue on a surface, it is important to keep a given translational cartesian position and *speed* to avoid the glue from clotting. However, it is not necessary to keep the desired *orientation* exactly. The second approach is also appropriate, if the desired trajectory is not known a priori and given by some sensor information, e.g. while *teaching* with a 6D input device.

Various approaches tackle the problem by an appropriate numerical treatment of a rank deficient Jacobian. Nakamura and Hanafusa [7] use a *damped least squares* algorithm, which leads to a poor performance of the manipulator near singularities. Senft, Hirzinger [12] and Nenchev [8, 14, 15] tackle singularities by utilizing the *adjoint Jacobian* matrix. Snell [13] uses a QR-decomposition to determine the rank-deficient subspace of the Jacobian. Siciliano [6] describes another algorithm, based on the *Jacobian transposed* matrix $\boldsymbol{J}^T$, which has been enhanced in [11].

In this paper, a constraint optimization approach as utilized for redundant robots [2, 9, 10], is used by reformulating the non–redundant inverse kinematics problem as a redundant problem. The latter idea was also sketched in [5]. With the constraint optimization approach, speed, acceleration and torque limits can be applied as inequality constraints and the acceptable deviation from the desired trajectory can be weighted for a certain cartesian subspace. In chapter 2 the redundant optimization problem will be formulated. In section 2.3, the approach will be improved by introducing an additional time scaling as optimization variable. In chapter 3 it is shown how the new method can be combined with an analytic solution of the inverse kinematic problem to enhance efficiency. Finally, chapter 4 shows some numerical results of the proposed algorithm.

2 Constraint optimization problem

2.1 Kinematic problem formulation

The forward kinematics of a robot manipulator is usually given by a set of nonlinear equations:

$$\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{q}) \tag{1}$$

where $\boldsymbol{x}$ is the 6 dimensional vector of cartesian coordinates, $\boldsymbol{f}$ is a 6 dimensional vector function mapping the n dimensional joint vector $\boldsymbol{q}$ to end effector coordinates. In the sequel, we are considering a non-redundant robot, i.e., $n = 6$ holds. Calculating the inverse of Equation (1)

$$\boldsymbol{q} = \boldsymbol{f}^{-1}(\boldsymbol{x}) \tag{2}$$

leads to high joint speeds and accelerations in the vicinity of a singularity. Below a numerical method is proposed to keep these quantities in their allowed limits. Deriving (1) leads to

$$d\boldsymbol{x} = \boldsymbol{J}(\boldsymbol{q})d\boldsymbol{q} \tag{3}$$

where $\boldsymbol{J}(\boldsymbol{q})$ is the manipulators Jacobian matrix. In a singular configuration $\boldsymbol{q}_s$ the Jacobian $\boldsymbol{J}(\boldsymbol{q}_s)$ looses rank and (3) can no longer be inverted.

2.2 Reformulation as a redundant problem

The mentioned problem shall now be reformulated as a redundant problem, which is solved by constraint optimization The vector equation (1) shall be split into a part which has to be fulfilled exactly (superscript *ex*) and a part which need to be fulfilled only approximately (superscript *ap*):

$$\boldsymbol{x}^{ex} = \boldsymbol{f}^{ex}(\boldsymbol{q}) \tag{4}$$

$$\boldsymbol{x}^{ap} = \boldsymbol{f}^{ap}(\boldsymbol{q}) \tag{5}$$

$$\dim(\boldsymbol{x}^{ex}) + \dim(\boldsymbol{x}^{ap}) = 6 \tag{6}$$

A good choice for splitting the two subspaces is the natural separation of the cartesian space into translational part (to be fulfilled exactly) and orientational part (to be fulfilled approximiately). In the sequel, vector $\boldsymbol{x}$ will be considered to be resorted, such that $\boldsymbol{x} = (\boldsymbol{x}^{ex^T} \boldsymbol{x}^{ap^T})^T$ holds.

The cartesian error $\boldsymbol{e}$ is defined as

$$\boldsymbol{e} = \begin{bmatrix} \boldsymbol{e}^{ex} \\ \boldsymbol{e}^{ap} \end{bmatrix} = \boldsymbol{x} - \boldsymbol{x}_d \tag{7}$$

where $\boldsymbol{x}$ is the current end effector position, defined by (1) and $\boldsymbol{x}_d$ is the desired cartesian position. The goal is that the error term $\boldsymbol{e}^{ex}$ vanishes and that the remaining part of the error term is as small as possible, i.e.,

$$\min_{\boldsymbol{q}} |\boldsymbol{e}^{ap}|^2 \quad \text{subject to} \quad \boldsymbol{e}^{ex} = \boldsymbol{0} \tag{8}$$

and subject to operational constraints, such as motor speed limits.

In figure 1 the most important symbols for the derivation below are explained. In particular, $\boldsymbol{q}_i = \boldsymbol{q}(t_i)$ are the unknown joint coordinates at sample instant t_i, which have to be determined, $\boldsymbol{q}_{i-1} = \boldsymbol{q}(t_{i-1})$ are the known joint coordinates from the last sample instant t_{i-1}, $\boldsymbol{q}_i^{(j)}$ are the joint coordinates of the j-th iteration in order to determine $\boldsymbol{q}_i$, $\Delta\boldsymbol{q}_i^{(j)}$ is the vector from the last sample instant to the j-th iteration point and $\delta\boldsymbol{q}_i^{(j)}$ is the vector from the last iteration point $j-1$ to the actual iteration point j.

The kinematic equations (1) are linearized around the last iterative solution vector $\boldsymbol{q}_i^{(j-1)}$, i.e.,

$$\boldsymbol{x}_i^{(j)} \approx \boldsymbol{f}(\boldsymbol{q}_i^{(j-1)}) + \boldsymbol{J}(\boldsymbol{q}_i^{(j-1)}) \cdot \delta\boldsymbol{q}_i^{(j)} \tag{9}$$

with $\delta\boldsymbol{q}_i^{(j)} = \boldsymbol{q}_i^{(j)} - \boldsymbol{q}_i^{(j-1)}$ and $\boldsymbol{q}_i^{(0)} = \boldsymbol{q}_{i-1}$. Since the constraints of the optimization problem are most naturally expressed as functions of $\Delta\boldsymbol{q}_i^{(j)}$, and not of $\delta\boldsymbol{q}_i^{(j)}$, the relationship

$$\delta\boldsymbol{q}_i^{(j)} = \boldsymbol{q}_{i-1} + \Delta\boldsymbol{q}_i^{(j)} - \boldsymbol{q}_i^{(j-1)} \tag{10}$$

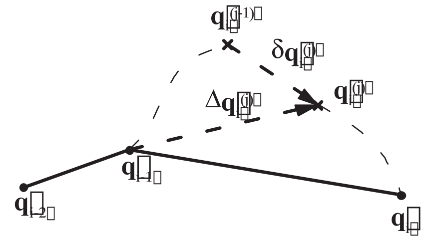

Figure 1. Definition of symbols for optimization problem

is used to express (9) as function of $\Delta\boldsymbol{q}_i^{(j)}$ leading to the following linear approximation of the cartesian error in the j-th iteration:

$$\boldsymbol{e}_i^{(j)} \approx \boldsymbol{J}(\boldsymbol{q}_i^{(j-1)})\Delta\boldsymbol{q}_i^{(j)} \; + \; \boldsymbol{b}_i^{(j)} - \boldsymbol{x}_{d,i} \tag{11}$$

with

$$\boldsymbol{b}_i^{(j)} = \boldsymbol{f}(\boldsymbol{q}_i^{(j-1)}) + \boldsymbol{J}(\boldsymbol{q}_i^{(j-1)})(\boldsymbol{q}_{i-1} - \boldsymbol{q}_i^{(j-1)}) \tag{12}$$

The most important operational constraints, such as constraints on position, velocity and acceleration of the joint coordinates can be expressed as

$$\Delta\boldsymbol{q}_{i,min} \leq \Delta\boldsymbol{q}_i^{(j)} \leq \Delta\boldsymbol{q}_{i,max} \tag{13}$$

with appropriately chosen minimal and maximal values $\Delta\boldsymbol{q}_{i,min}$ and $\Delta\boldsymbol{q}_{i,max}$, respectively. For example, limits on the joint speeds

$$\dot{\boldsymbol{q}}_{min} \leq \dot{\boldsymbol{q}}(t) \leq \dot{\boldsymbol{q}}_{max} \tag{14}$$

can be discretized

$$\dot{\boldsymbol{q}}_i^{(j)} \approx \frac{\boldsymbol{q}_i^{(j)} - \boldsymbol{q}_{i-1}}{t_i - t_{i-1}} = \frac{\Delta\boldsymbol{q}_i^{(j)}}{\Delta t_i} \tag{15}$$

leading to the following structure of (13)

$$\Delta t_i \cdot \dot{\boldsymbol{q}}_{min} \leq \Delta\boldsymbol{q}_i^{(j)} \leq \Delta t_i \cdot \dot{\boldsymbol{q}}_{max} \tag{16}$$

We are now in the position to formulate the first version of the optimization problem in a formal manner, by splitting (11) according to the previous definitions and by utilizing (13):

$$\min_{\Delta\boldsymbol{q}_i^{(j)}} \left|\boldsymbol{e}_i^{ap,(j)}\right|^2 \quad \text{subject to} \tag{17}$$

$$\boldsymbol{e}_i^{ex,(j)} = \boldsymbol{0}, \quad \Delta\boldsymbol{q}_{i,min} \leq \Delta\boldsymbol{q}_i^{(j)} \leq \Delta\boldsymbol{q}_{i,max} \tag{18}$$

This is a linear least squares problem with linear equality and inequality constraints. If the Jacobian $\boldsymbol{J}(\boldsymbol{q}_i^{(j-1)})$ has full rank, a *unique* solution to this optimization problem exists. If the Jacobian is rank deficient, i.e., the manipulator is in a singular configuration, there is an infinite number of solutions. The solution can be made unique by requiring that $|\Delta \boldsymbol{q}_i^{(j)}|$ is minimized additionally in this situation, i.e., the changes of the joint coordinates with respect to the last sample instant shall be as small as possible. Standard numerical algorithms exist to solve this special convex optimization problem in a finite number of steps in a reliable way, see e.g. [4, 1] for details.

By successive linearizations around the last solution point, several of these optimization problems are stated and solved until the solution of the original nonlinear optimization problem (8) is approximated sufficiently enough.

In a well conditioned configuration, this formulation leads to the same result as the analytic solution (2), since effectively the nonlinear equations (2) are solved. In ill–conditioned areas, only the requested exact equations are fulfilled and the others are approximated as a secondary subtask, in order to fulfill the operational constraints.

It turns out that the solution of the discussed optimization problem for successive points often leads to undesired oscillations in the joint coordinates. By requiring that also the time derivative of the cartesian error, i.e., $\dot{\boldsymbol{e}}^{ap}$, shall be minimized, these oscillations can be effectively removed. With

$$\begin{aligned} \dot{\boldsymbol{e}} &= \dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_d & (19)\\ &= \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} - \dot{\boldsymbol{x}}_d & (20) \end{aligned}$$

the derivative of the cartesian error $\dot{\boldsymbol{e}}_i^{(j)}$ in the j-th iteration at the i-th time instant can be approximated by

$$\dot{\boldsymbol{e}}_i^{(j)} \approx \boldsymbol{J}(\boldsymbol{q}_i^{(j-1)}) \frac{\Delta \boldsymbol{q}_i^{(j)}}{\Delta t_i} - \frac{\Delta \boldsymbol{x}_{d,i}}{\Delta t_i} \qquad (21)$$

with

$$\begin{aligned} \Delta t_i &= t_i - t_{i-1} & (22)\\ \Delta \boldsymbol{x}_{d,i} &= \boldsymbol{x}_d(t_i) - \boldsymbol{x}_d(t_{i-1}) & (23) \end{aligned}$$

The two criterias are weighted with respect to each other with weighting matrices $\boldsymbol{\alpha}$, $\Delta t_i \boldsymbol{\beta}$, respectively, i.e., the following criteria is minimized:

$$\min_{\Delta \boldsymbol{q}_i^{(j)}} \left|\boldsymbol{\alpha} \boldsymbol{e}_i^{ap,(j)}\right|^2 + \left|\Delta t_i \boldsymbol{\beta} \dot{\boldsymbol{e}}_i^{ap,(j)}\right|^2 \qquad (24)$$

Collecting all formulas together, finally leads to the following linear least squares problem formulation: Find a solution vector $\Delta \boldsymbol{q}_i^{(j)}$ such that the criteria

$$\min_{\Delta \boldsymbol{q}_i^{(j)}} \left| \begin{bmatrix} \boldsymbol{\alpha} \boldsymbol{J}^{ap}(\boldsymbol{q}_i^{(j-1)}) \\ \boldsymbol{\beta} \boldsymbol{J}^{ap}(\boldsymbol{q}_i^{(j-1)}) \end{bmatrix} \Delta \boldsymbol{q}_i^{(j)} - \begin{bmatrix} \boldsymbol{\alpha}(\boldsymbol{x}_{d,i}^{ap} - \boldsymbol{b}_i^{ap,(j)}) \\ \boldsymbol{\beta} \Delta \boldsymbol{x}_{d,i}^{ap} \end{bmatrix} \right|^2 \qquad (25)$$

is minimized subject to the linear equality constraints which have to be fulfilled exactly

$$\boldsymbol{J}^{ex}(\boldsymbol{q}_i^{(j-1)}) \Delta \boldsymbol{q}_i^{(j)} + \boldsymbol{b}_i^{ex,(j)} - \boldsymbol{x}_{d,i}^{ex} = \boldsymbol{0} \qquad (26)$$

and subject to the linear inequality constraints

$$\begin{aligned} \Delta \boldsymbol{q}_i^{(j)} &\leq \Delta \boldsymbol{q}_{i,max}(\boldsymbol{q}_{max}, \dot{\boldsymbol{q}}_{max}, \ddot{\boldsymbol{q}}_{max}) & (27)\\ \Delta \boldsymbol{q}_i^{(j)} &\geq \Delta \boldsymbol{q}_{i,min}(\boldsymbol{q}_{min}, \dot{\boldsymbol{q}}_{min}, \ddot{\boldsymbol{q}}_{min}) & (28) \end{aligned}$$

The main advantage of the proposed algorithm is, that the optimizer selects $\boldsymbol{x} = \boldsymbol{x}_d$, if this is possible. As soon as the desired trajectory exceeds the physical limits of the manipulator, a cartesian error in a specified subspace occurs. This error is made as small as possible under the constraint that the physical limits are not exceeded.

In other words: The proposed algorithm describes a redundant manipulator with m^{ex} degrees of freedom. The primary subtask is to fulfill the respective constraints exactly, the secondary subtask is to fulfill the m^{ap} error equations and the m^{ap} derivative error equations approximately.

The optimization problem can be further improved by adding additional criterias, e.g., (a) the maximum cartesian error can be introduced as additional inequality constraint to guarantee that the cartesian error is limited, (b) the maximum acceleration constraint can be replaced by maximum torque limits of the motors. Via the dynamic equations of the robot, these torque limits can be transformed into a set of linear inequality constraints.

2.3 Generalizing the constraint optimization approach by time scaling

The proposed structure of the optimization problem (25) can be generalized by including a time scaling of the *desired* trajectory to reduce the cartesian speed in the vicinity of a singularity. With this time parameterization, the same handling of singularities is possible, than in [3]. The trajectory shall be a function of time

$$\boldsymbol{x}_d = \boldsymbol{g}(t) \qquad (29)$$

So the problem may be linearized similarily as in the previous chapter around the last time step $i-1$.

$$t_i = t_{i-1} + \Delta t_i \qquad (30)$$

$$\boldsymbol{x}_{d,i} \approx \boldsymbol{g}(t_{i-1}) + \frac{d}{dt}\boldsymbol{g}(t_{i-1})\Delta t_i \qquad (31)$$

where Δt_i is the sample instant. In order that the physical limits can be easier fulfilled, the desired trajectory is modified in such a way that the robot moves more slowely along the desired geometric path. This behaviour can be reached by scaling the time parameter of the desired trajectory, leading to a modified desired trajectory $\boldsymbol{x}_d^*(t)$ with

$$\begin{aligned} \boldsymbol{x}_{d,i}^* &\approx \boldsymbol{g}(t_{i-1}) + \frac{d}{dt}\boldsymbol{g}(t_{i-1})(1-\vartheta)\Delta t_i & (32)\\ &\approx \boldsymbol{x}_{d,i-1} + \Delta \boldsymbol{x}_{d,i} \cdot (1-\vartheta) & (33) \end{aligned}$$

where $\vartheta = 0$ characterizes the desired trajectory, $0 < \vartheta < 1$ characterizes a slower desired trajectory and $\vartheta = 1$ leads to a complete stop. ϑ is used as additional unknown variable which is determined in the optimization process. Since the desired trajectory should be reached, an additional criteria with weighting factor γ has to be introduced which minimizes $|\gamma\vartheta|^2$ in order that ϑ is as close to zero as possible. Incorporating this criteria into (25), replacing $\boldsymbol{x}_{d,i}$ by $\boldsymbol{x}^*_{d,i}$ and utilizing (33) leads to the following criteria which has to be minimized:

$$\min \left\| \begin{bmatrix} \boldsymbol{\alpha}\boldsymbol{J}^{ap}(\boldsymbol{q}_i^{(j-1)}) & \boldsymbol{\alpha}\Delta\boldsymbol{x}_{d,i}^{ap} \\ \boldsymbol{\beta}\boldsymbol{J}^{ap}(\boldsymbol{q}_i^{(j-1)}) & \boldsymbol{0} \\ \boldsymbol{0} & \gamma \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{q}_i^{(j)} \\ \vartheta \end{bmatrix} - \begin{bmatrix} \boldsymbol{\alpha}(\boldsymbol{x}_{d,i}^{ap} - \boldsymbol{b}_i^{ap,(j)}) \\ \boldsymbol{\beta}\Delta\boldsymbol{x}_{d,i}^{ap} \\ 0 \end{bmatrix} \right\|^2 \tag{34}$$

The augmented problem in (34) leads to the same problem than (25) by fixing $\vartheta = 0$. Choosing $\vartheta \in [0, 1]$ and $m^{ap} = 0$, one will end up with a constraint optimization problem, which solves the problem of *untimed trajectories* in [3].

The new introduced optimization variable ϑ can be interpreted as *task dependent dexterity measure*, which complies to the given limits.

3 Considerations about switching inverse kinematic algorithms

For most industrial robots, the inverse kinematic problem (2) outside of a singularity can be solved analytically. Such an analytic solution is about 10–20 times faster as the solution of the optimization problem (25). The analytic solution only lacks the treatment of the singularity problem, or more precisely, the prevention and prediction of exceeding physical limits of the manipulator. So, to solve the singularity problem, one should think about combining the algorithms: Using the efficient analytic algorithm under all feasible circumstances and the new constraint optimization approach only when necessary. Such an approach requires appropriate switching conditions between the two algorithms.

One possibility could be to detect a rank deficiency of the manipulators Jacobian in equation (3), since this characterizes the singular positions of the manipulator. However, the physical limits of the manipulator are already violated when the Jacobian is still regular, but ill-conditioned.

So the switching conditions have to be formulated along the physical limits, formulated in equation (13). If the result $\boldsymbol{q}_i^{\boldsymbol{a}}$ of the analytic solution does not comply with the inequality constraints of (13), the iterative solution is used.

$$\Delta\boldsymbol{q}_i^{\boldsymbol{a}} = \boldsymbol{q}_i^{\boldsymbol{a}} - \boldsymbol{q}_{i-1}^{\boldsymbol{a}} \quad \notin [\Delta\boldsymbol{q}_{i,min}, \Delta\boldsymbol{q}_{i,max}] \tag{35}$$

Similarly, a switching from the constraint optimization approach to the analytic solution can take place, if the solution of the optimization problem is no longer on the border of the inequality constraints (= no active inequality constraints).

4 Numerical Results

The proposed algorithm of (25) has been tested with a KUKA industrial robot (see figure 2). The robot can be teached with a sensor-device, and driven through singularities. Figures 3 to 18 show two experiments, how the

Figure 2. KUKA standard industry robot

proposed iterative algorithm overrides the analytic solution while traveling through the typical axis 5 singularity. The cartesian positions have been chosen to be fulfilled exactly ($m^{ex} = 3$), and the orientations approximately ($m^{ap} = 3$).

As shortly discussed at the end of chapter 2.2, an upper limit for the cartesian error has been introduced. This leads to an implicit feasibility test of the trajectory. In some practical experiments it turned out that accepting about 3 degrees of deviation is enough for the teaching application, commanded by a 6D input device. The optimization problem with 2 iterations for one sampling instant was solved in about 3 milliseconds on a Pentium 100 MHZ machine. Some performance improvements are possible, taking the special structure of the formulated problem into account.

Example 1 (figures 3, 5-11) shows a motion in the vicinity of the axis five singularity. The start and goal configuration is shown by figure3, the motion travels from start to goal configuration, and back again. The plots of figures 5-10 show the motion of a single axis, in the first row, numerical and analytical solution is plotted, in the second row, the difference between analytic and numeric solution is shown, in the third row, the speed of analytic and numeric and in the forth the accelerations are plotted. In figure 5, it is important to observe the orientational deviation is executed by a motion of the first axis, which would be in rest by the analytic solution. Axis 4 and Axis 6 (figures 8 and 10) show the motion beyond the limits close to the singularity. The analytic solution exceeds the limits, and the numeric solutions sticks on the limits of acceleration and speed. The cartesian error development during the trajectory shows 11. In the upper plot the rotational error is displayed ($\boldsymbol{e^{ap}}$), which is growing due to the physical limits of the manipulator and in the lower plot, the time derivative of the error $\Delta \boldsymbol{e^{ap}}$ is displayed.

Example 2 (figures 4, 12-18) shows a motion perpendicular to the singulating axis 5. The analytic solution steps in the first step of the trajectory by 90 Degree ($\pi/2$ rad) in axis 4 and axis 6 (figures 8 and 10).

This example shows the performance of the approach, which allows intuitive guidance of the manipulator with sensor devices. The user is allowed to command even infeasible trajectories and the manipulator follows with an accepted deviation.

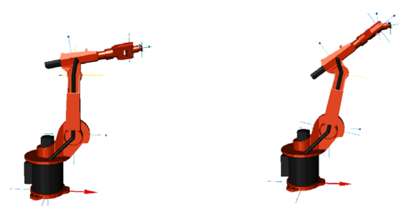

Figure 3. Start and goal configuration of the first example, traveling in the vicinity of the axis 5 singularity, the motion is from start to goal configuration and back again.

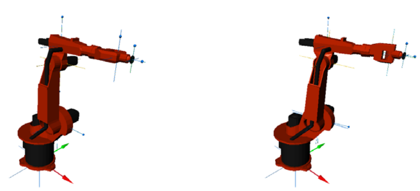

Figure 4. Start and goal configuration of second example. The manipulator is commanded a trajectory perpendicular to the axis 5 singularity.

5 Conclusions

The usage of constraint optimization algorithms turns out to be very useful when solving the singularity problem of non–redundant robots. The main advantage of this formulation lies in the explicit consideration of physical limits

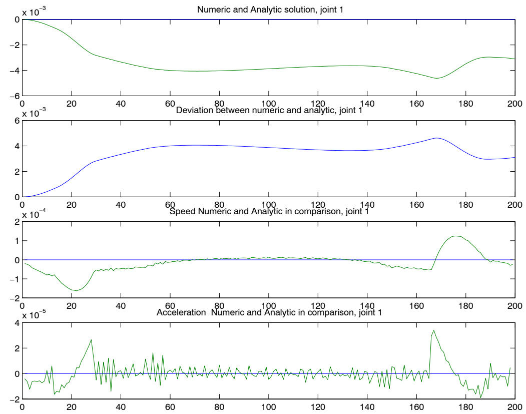

Figure 5. Axis 1 of first example, the deviation is executed with motion of the first axis.

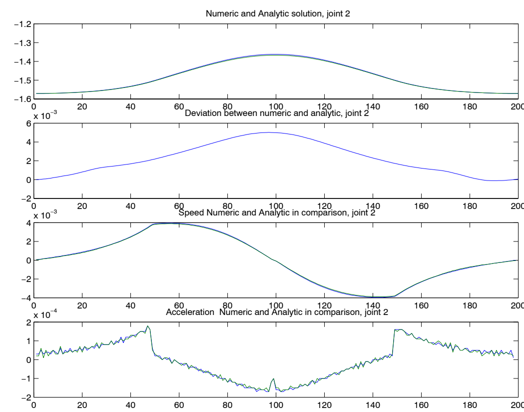

Figure 6. Axis 2 of first example

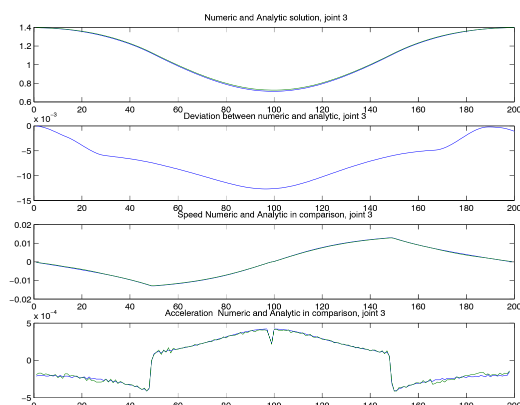

Figure 7. Axis 3 of first example

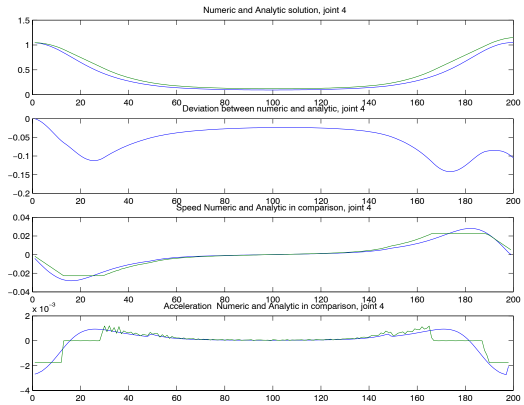

Figure 8. Axis 4 of first example, within the speed and acceleration plots, one can observe the sticking of the algorithm on the physical limits.

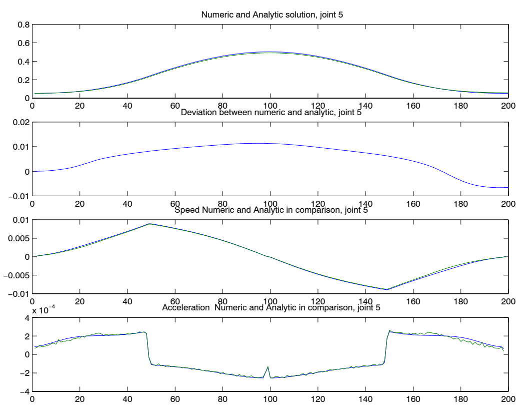

Figure 9. Axis 5 of first example

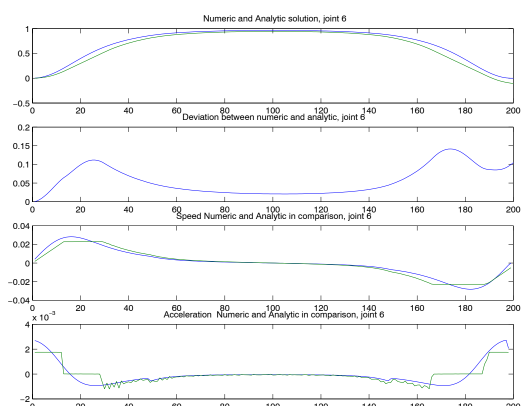

Figure 10. Axis 6 of first example, within the speed and acceleration plots, one can observe the sticking of the algorithm on the physical limits.

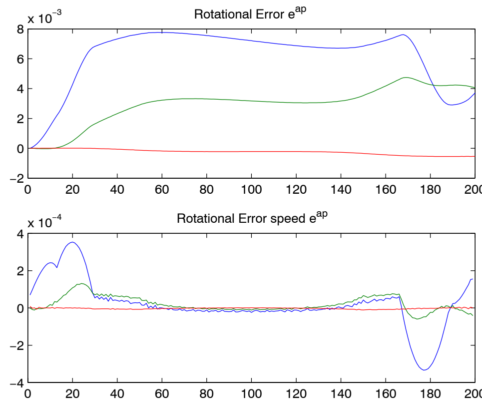

Figure 11. Cartesian error development of first example. The upper plot shows the error development of the approximate error e_i^{ap} (rotational) and the lower plot shows the development of the error derivative Δe_i^{ap}

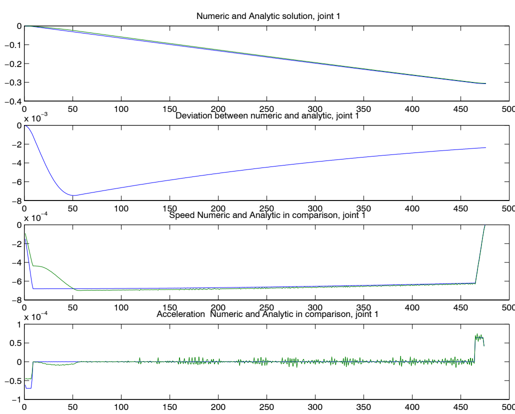

Figure 12. Axis 1 of second example

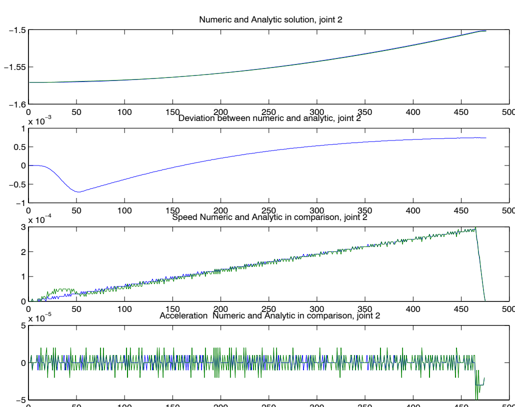

Figure 13. Axis 2 of second example

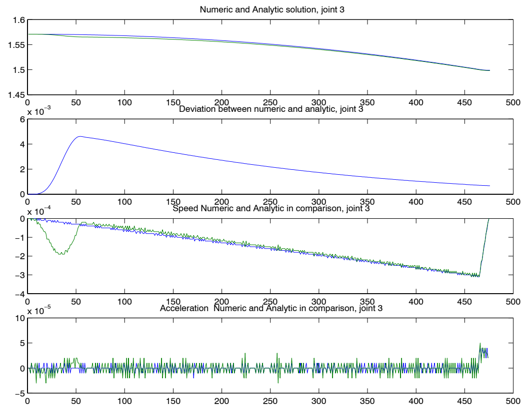

Figure 14. Axis 3 of second example

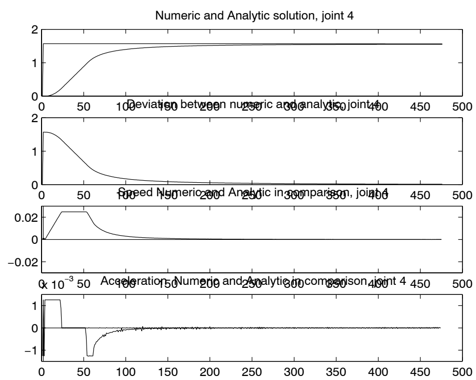

Figure 15. Axis 4 of second example, here the analytic solution steps within one time step about $\pi/2$ (90 Degree), the iterative solution shows the physical feasible motion to execute the trajectory

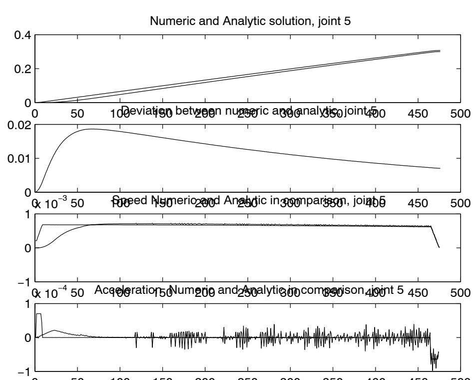

Figure 16. Axis 5 of second example

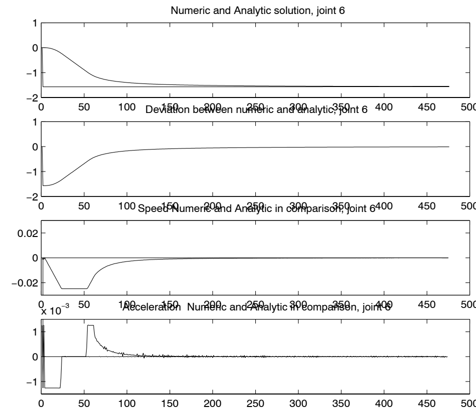

Figure 17. Axis 6 of second example, here the analytic solution steps within one time step about $\pi/2$ (90 Degree), the iterative solution shows the physical feasible motion to execute the trajectory

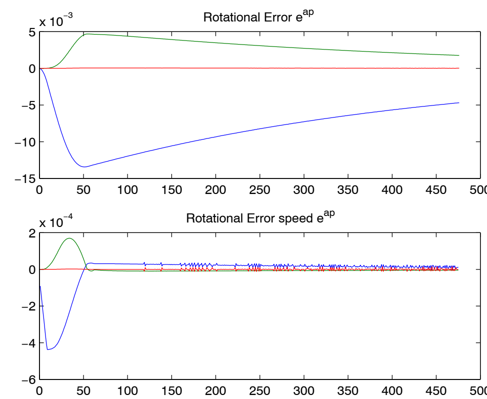

Figure 18. Cartesian error of example 2, The upper plot shows the error development of the approximate error $\boldsymbol{e_i^{ap}}$ (rotational) and the lower plot shows the development of the error derivative $\Delta\boldsymbol{e_i^{ap}}$

within the optimization problem. This is a key to the solution of the problem, since cartesian trajectory tracking can now be executed close to the physical possibilities of the mechanism.

This kind of solution to the singularity problem solves one class of singularity problems. It can be applied especially for teaching with sensor-devices, like the "DLR–SpaceMouse" and for tracking of a priori known trajectories, where it is more important to track cartesian speed than to follow exactly the trajectory orientation, e.g. for applications like dispensing glue with a robot. The algorithm has been generalized by augmenting with an additional timing parameter ϑ, which solves either the problem of exact traveling on trajectories by slowing down or by combination of both algorithms. The presented algorithm has been shown to work on a KUKA industrial robot, and will be applied to redundant systems, like the 7–dof "DLR-Lightweight-Robot".

References

[1] A. Björk. *Numerical Methods for Least Squares Problems*. SIAM, 1996.

[2] F.-T. Cheng, T. Chen, and Y.-Y. Sun. Resolving manipulator redundancy under inequality constraints. *IEEE Transaction on Robotics and Automation*, 10(1):65–71, Feb 1994.

[3] J. Kieffer. Manipulator inverse kinematics for untimed end-effector trajectories with ordinary singularities. *Int. J. Robotics Research*, 11(3):225–237, 1992.

[4] C. Lawson and R. Hanson. *Solving Least Squares Problems*. Prentice Hall, 1974.

[5] J. Lenarčič. Alternative computational scheme of manipulator inverse kinmatics. In *IEEE International Conference on Robotics and Automation*, pages 3235–3240, Leuven, Belgium, May 1998.

[6] L.Sciavicco and B.Siciliano. *Modelling and Control of Robot Manipulators*. ISBN 0-07-057217-8. McGraw Hill Companies, Inc., 1996.

[7] Y. Nakamura and H. Hanafusa. Inverse kinematic solutions with singularity robustness for robot manipulator control. *Journal of Dynamic Systems, Measurement ,and Control*, 108:163–171, September 1986.

[8] D. Nenchev, V. Senft, and G. Hirzinger. Two approaches to singularity-consistent motion of nonredundant robotic mechanisms. In *International Conference on Robotics and Automation*, pages 1883–1890, Minneapolis, Minnesota, April 1996. IEEE.

[9] K. C. Park, P. H. Chang, and J. K. Salisbury. A unified approach for local resolution of kinematic reundancy with inequality constraints and its application to nuclear power plant. In *Procieedings of the 1997 IEEE International Conference on Robotics and Automation*, pages 766–773, Albuquerque, New Mexico, April 1997.

[10] M. Schlemmer. *Eine SQP-Rückwärtskinematik zur interaktiven, semi-autonomen Bahführung kinematisch redundanter Manipulatoren*. PhD thesis, Gerhard-Mercator-Universität, Gesamthochschule Duisburg, submitted to PhD examination board December 1997.

[11] G. Schreiber and G. Hirzinger. Singularity consistent inverse kinematics by enhancing the jacobian transpose. In J. Lenarčič and M. L. Husty, editors, *Advances in Robot Kinematics: Analysis and Control*, pages 475–483, Strobl/Wolfgangsee, 1998. Kluwer Academic Publishers, Dordrecht.

[12] V. Senft and G. Hirzinger. Redundant motions of non redundant robots - a new approach to singularity treatment. In *International Conference on Robotics and Automation*, pages 1553–1558, Nagoya, Japan, May 1995. IEEE.

[13] J.-E. Snell. *Method for controlling the movements of an industrial robot at and near singularities*. European Patent Application, Publication number: 0 672 507 A1, Application number: 95102241.7, Date of filing: 18. 2. 1995, Applicant: ABB.

[14] Y. Tsumaki, S. Kotera, D. Nenchev, and M. Uchiyama. Singularity–consistent inverse kinmatics of a 6 dof manipulator with a non-spherical wrist. In *International Conference on Robotics and Automation*, pages 2980–2985, Albuquerque, New Mexico, April 1997. IEEE.

[15] Y. Tsumaki, D. Nenchev, and M. Uchiyama. Experimental teleoperation of a nonredundant slave arm at and around singularities. In *International Conference on Robotics and Automation*, pages 385–392, Minneapolis, Minnesota, April 1996. IEEE.