PAMM · Proc. Appl. Math. Mech. **14**, 75 – 76 (2014) / **DOI** 10.1002/pamm.201410025

Time Optimal Path Planning for Industrial Robots: A Dynamic Programming Approach Considering Torque Derivative and Jerk Constraints

Matthias Oberherber[1,*], **Hubert Gattringer**[1], and **Klemens Springer**[1]

[1] Institute for Robotics, Johannes Kepler University, Altenbergerstraße 69, 4040 Linz, Austria

This paper presents a dynamic programming approach for calculating time optimal trajectories for industrial robots, subject to various physical constraints. In addition to path velocity, motor torque, joint velocity and acceleration constraints, the present contribution also shows how to deal with torque derivative and joint jerk limitations. First a Cartesian path for the endeffector is defined by splines using Bernstein polynomials as basis functions and is parameterized via a scalar path parameter. In order to compute the belonging quantities in configuration space, inverse kinematics is solved numerically. Using this and in combination with the dynamical model, joint torques as well as their derivatives can be constrained. For that purpose the equations of motion are calculated with the help of the Projection Equation. As a consequence of the used optimization problem formulation, the dynamical model as well as the restrictions have to be transformed to path parameter space. Due to the additional consideration of jerk and torque derivative constraints, the phase plane is expanded to a phase space. The parameterized restrictions lead to feasible regions in this space, in which the optimal solution is sought. Result of the optimization is the time behavior of the path parameter and subsequently the feed forward torques for the optimal motion on the spatial path defined by previously mentioned splines. Simulation results as well as experimental results for a three axes industrial robot are presented.

1 Spatial Path Planning and Dynamic Modeling

A geometric path as well as the robots mathematical model are the foundations for the optimization. There are many different ways to define geometric paths, like line - curve combinations, polynomials, or splines. Latter ones are used in the present paper to define a path $\mathbf{z}_E(\sigma) = \sum_j \mathbf{d}_j R_j^d(\sigma)$ in world coordinates as a function of a scalar path parameter σ. R_j^d are the Bernstein polynomials of degree d representing the basis functions. The control points $\mathbf{d}_j$ determine the path's shape, shown in Fig. 1. The inverse kinematics provides the joint angles $\mathbf{q}(\sigma) = \mathbf{f}(\mathbf{z}_E(\sigma))$ as a function of the endeffector coordinates. A dynamic model of the robot is necessary on the one hand to be able to simulate the robot system and on the other hand to compute the feed-forward torques used for optimization and control. Its calculation is done with the help of the Projection Equation (see [4]) , resulting in the equations of motion $\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Q}$.

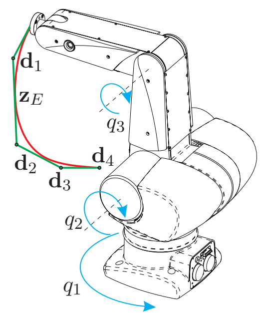

Fig. 1: 3DOF Robot

2 Optimal Motion Along Defined Paths

This section shows an approach to calculate an optimal movement on a geometrical defined path. The cost functional in Eq. (1) that has to be minimized for a time optimal solution depends on the unknown cycle time t_E. With $\dot{\sigma} = \frac{d\sigma}{dt}$ and the substitution $z = \dot{\sigma}^2$, Eq. (1) can be rewritten as a function of the path parameter σ. Similarly, the constraints can be transformed into the parameter range by parametrizing the equations of motion, its derivatives with respect to time and the motion quantities (Eq. (2)). Afterwards the optimization can be executed in the path parameter space.

Concerning this subject there exist several methods but especially the works of Pfeiffer et al. [2] and Bobrow et al. [3] provide the basis for effective algorithms. While Constantinescu et al. [6] show how to deal with jerk and torque rate constraints using a method based on switching points, the presented paper relies on a Dynamic Programming Approach, based on Bellman's Optimality Principle [1].

$$W = \min_{t_E \in \mathbb{R}} \int_0^{t_E} 1 dt \qquad (1)$$
$$s.t. \; [|\mathbf{Q}| \; |\dot{\mathbf{Q}}|](t) \le [\mathbf{Q}, \; \dot{\mathbf{Q}}]_{max}$$
$$[|\dot{\mathbf{q}}|, \; |\ddot{\mathbf{q}}|, \; |\dddot{\mathbf{q}}|, \; v_E](t) \le [\,\dot{\mathbf{q}}, \; \ddot{\mathbf{q}}, \; \dddot{\mathbf{q}}, \; v_E]_{max}$$

$$W = \min_{z \in \mathbb{R}} \int_0^{\sigma_E} \frac{1}{\sqrt{z}} d\sigma \qquad (2)$$
$$s.t. \; [|\mathbf{Q}| \; |\dot{\mathbf{Q}}|](\sigma) \le [\mathbf{Q}, \; \dot{\mathbf{Q}}]_{max}$$
$$[|\dot{\mathbf{q}}|, \; |\ddot{\mathbf{q}}|, \; |\dddot{\mathbf{q}}|, \; v_E](\sigma) \le [\dot{\mathbf{q}}, \; \ddot{\mathbf{q}}, \; \dddot{\mathbf{q}}, \; v_E]_{max}$$

* Corresponding author: e-mail matthias.oberherber@jku.at, phone +43 732 2468 6495, fax +43 732 2468 6492

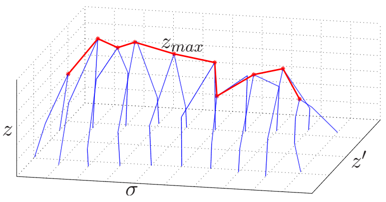

Fig. 2: Velocity limit curve

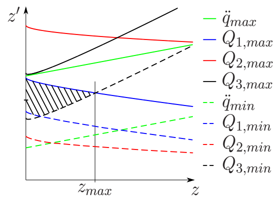

Fig. 3: $[z\ z']$ plane

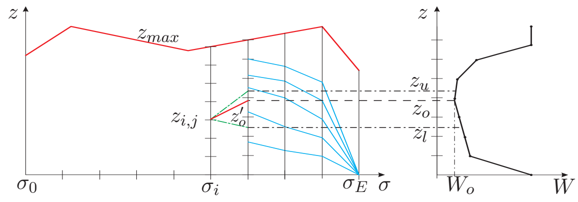

Fig. 4: Dynamic Programming Approach

For that purpose, path ($\sigma = \sigma_0, \sigma_1 \dots \sigma_n$), velocity ($z = 0, z_1, \dots z_m = z_{max}$) and the cost functional W are discretized and the problem is handled in the discrete grid of the $[\sigma, z]$ plane. In this plane an optimal velocity trend minimizing the cost functional W is calculated by evaluating optimal gradients $z' = dz/d\sigma$. The velocity limit curve in Fig. 2 shows the trend of maximal allowed velocities z_{max} provided by the feasible region at every path point σ_i ($i = 0..n$). This feasible region (hatched area in Fig. 3) is bounded by velocity, torque and acceleration constraints. The limit curve is the basis of the procedure starting at the end of the path by evaluating a search region in the cost function $[z_l \dots z_u]$ with the help of the feasible region at this point. The optimal gradient at a discrete point $z_{i,j}$ ($j = 0 \dots m$) can be calculated with the location z_o of the cost functions minimum W_o to $z'_o = (z_o - z_{i,j})/\Delta\sigma$. Jerk and torque derivative constraints cause a feasible region in the $[z', z'']$ plane similar to Fig. 3. The optimal gradient has to be checked for feasibility in this plane and adapted if necessary. A gradient z'_o fulfilling all restrictions is stored for this point (i, j) and the cost function is adapted to $W_{i,j} = W_o + \Delta\sigma/\sqrt{z_{i,j}}$. After completing this procedure for every point in the discrete $[\sigma, z]$ plane, the optimal velocity trend can be calculated iteratively $z_{i+1,opt} = z_{i,opt} + z'_{i,opt}\Delta\sigma$. With z_{opt} on hand the time vector is obtained by $t_i = \sum_{k=1}^{i} \Delta t_k$, with $\Delta t_k = \Delta\sigma / z_{k,opt}$. This leads to a path parameter trend over time $\sigma(t)$ representing the optimization result.

3 Results

For validation purposes the algorithms are implemented on a Catalyst CRS robot, shown in Fig. 1. The optimization results for a straight line in space ($l \approx 0.74$m) are provided by Fig. 5 in terms of an optimal velocity profile z_{opt} and $\sigma(t)$. Measurements of the motion quantities are shown in Fig. 6, and of the appropriate torque and torque derivatives in Fig. 7. One can see that most of the time the velocity restrictions are active. In the acceleration and deceleration phases jerk and acceleration constraints are active. The cycle time for this experiment results to $t_E \approx 1.1$s while a cycle time of $t_E \approx 0.99$s can be reached by neglecting jerk constraints. The calculation time strongly depends on the chosen discretization and the duration of active jerk constraints. For this example the required calculation time is $t_{CPU} \approx 35$s using not optimized Matlab code.

A combination of the presented paper and the work in [5] offers the possibility to calculate time (semi-) optimal point to point motions considering jerk and torque derivative restrictions.

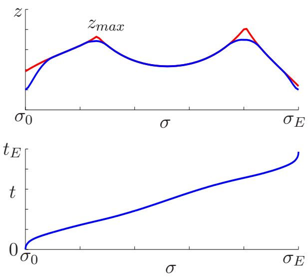

Fig. 5: Optimization results

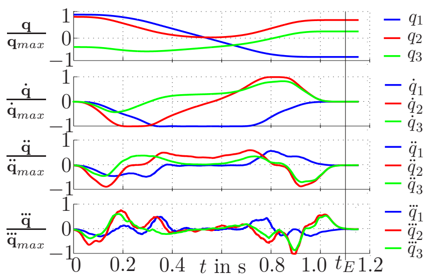

Fig. 6: Measurements of motion quantities

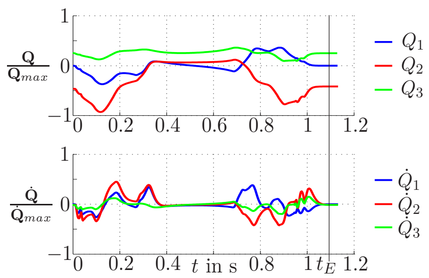

Fig. 7: Torque Measurements

Acknowledgements This work has been supported by the Austrian COMET-K2 program of the Linz Center of Mechatronics (LCM), and was funded by the Austrian federal government and the federal state of Upper Austria.

References

[1] R. E. Bellman, S. E. Dreyfus, Applied Dynamic Programming, (Princeton University Press, Princeton, New Jersey, 1962).
[2] F. Pfeiffer, R. Johanni, A Concept for Manipulator Trajectory Planning, IEEE J. Rob. Aut. **3**, pp. 115-123 (1987).
[3] J. E. Bobrow, S. Dubowsky, J. S. Gibson, Time-Optimal Control of Robotic Manipulators Along Specified Paths, Int. J. Rob. Res. **4**, pp. 3-17 (1985).
[4] H. Bremer, Elastic Multibody Dynamics: A Direct Ritz Approach (Springer-Verlag, Heidelberg, 2008).
[5] H. Gattringer, M. Oberherber, K. Springer, Extending continuous path trajectories to point-to-point trajectories by varying intermediate points, Int. J. Mech. Cont. **15**, pp. 35-43 (2014).
[6] D. Constantinescu, E. A. Croft, Smooth and Time-Optimal Trajectory Planning for Industrial Manipulators Along Specified Paths, J. Rob. Sys. **17**, pp. 233-249 (2000).

