Proceedings of the RAAD 2014
23rd International Conference on Robotics in Alpe-Adria-Danube Region
September 3-5, 2014, Smolenice Castle, Slovakia

A Real-Time Nearly Time-Optimal Point-To-Point Trajectory Planning Method Using Dynamic Movement Primitives

Klemens Springer, Hubert Gattringer and Christoph Stöger
Institute for Robotics
Johannes Kepler University Linz
Linz, Austria
Email: {klemens.springer, hubert.gattringer, christoph.stoeger}@jku.at

***Abstract*—This paper focuses on highly efficient real-time capable time-optimal trajectory planning for point-to-point motions. A method based on the concept of dynamic movement primitives is developed. Thereby an offline generated time-optimal point-to-point trajectory considering nonlinear physical constraints serves as reference for a movement primitive. Through variation of the goal position in a small range around the reference target, new nearly time-optimal point-to-point trajectories are obtained. For the required reconsideration of the physical constraints, a strategy is derived from a common minimum-time optimization problem formulation. Finally a comparison between this and existing real-time capable time-optimal trajectory planning methods is drawn using a six degree-of-freedom serial robot.**

I. Introduction

Time-optimal trajectory planning is a very important topic in research nowadays. Not least because automation industry has a high interest in increasing their production rates as much as possible. Obviously the most crucial part here is the demand for real-time computation of such trajectories. This is necessary in the field of vision based pick-and-place tasks or other processes with dynamically changing target positions for example.

The motion planning problem is divided into a geometric path planning and a trajectory planning stage often, whereby only in the latter one the robot dynamics as well as constraints are taken into account. Most of these methods exploit that motion on a geometric path, defined in the first stage, can be described by a single path coordinate σ, see [1], [2]. Following the resulting optimization problem can be solved efficiently using direct methods for example, see [3], [4]. In comparison to these path-constrained time-optimal motions (PCTOM), point-to-point time-optimal (PTPTO) trajectories utilize the robots whole workspace, that can result in a noticeable decrease of movement duration. Typically, PTPTO motion problems are solved using a bisection algorithm ([5], [6], [7]). Thereby a feasibility problem is solved in order to determine the minimum number of required discrete samples N_{min} for minimal motion time. Whereas real-time capable algorithms for PCTOM are already developed, see [8], [9], real-time algorithms for PTPTO trajectories with respect to nonlinear physical constraints do barely exist to the authors best knowledge.

This paper introduces a novel strategy to overcome the lack of current algorithms and allows for nearly time-optimal point-to-point motions with a small variation around the specified target position. An offline generated PTPTO trajectory is provided as a basis for this concept. This motion is transformed to a dynamic movement primitive (DMP) by utilizing a learning algorithm ([10], [11]). The resulting DMP serves as task-specific reference, similar to [12]. The advantage of this type of path description is the possibility to vary the goal position as well as the movement speed in real-time without a recomputation of the whole trajectory ([13]). As a consequence of the optimality of the learned motion it has been originated from, a violation of the constraints when moving on the adjusted trajectory is obvious. Hence the trajectory is scaled in time, whereby the required scaling is estimated by using the linearized inequality bounds from the original optimization problem formulation. An optimality as well as complexity analysis show the advantageous behavior of this method. For the verification of the resulting nearly time-optimal trajectories concerning optimality, an application example is given using a six degree-of-freedom (DoF) serial robot.

The paper is organized as follows: The time-optimal problem formulation and the used software package for obtaining the solution are dealt with in Section II. Section III gives a brief introduction to dynamic movement primitives and their properties as well as its application to this problem formulation. Also the time-scaling estimation and a computation analysis are done in this section. The application of the proposed trajectory planning method to a six DoF robot is explained in IV. Results of the time-optimal trajectory planning algorithm and its comparison to existing methods are discussed in V.

A. Notation

Throughout the whole paper italic letters are used for scalars a, bold lowercase ones for vectors $\mathbf{b}$ and matrices are written with bold uppercase letters $\mathbf{C}$.

II. Time-optimal path planning

In this section the calculation of the PTPTO reference trajectory is explained in general in combination with the introduction of the used software package.

For the computation of this trajectory there exist several solution approaches from indirect methods to direct ones using spline path or system input parametrizations. The latter one is used here in combination with the optimization software package *MUSCOD-II* ([14]) to obtain the optimal solution,

978-1-4799-6798-8/14/$31.00 ©2014 IEEE

that serves as a basis for dynamic movement primitives later on. *MUSCOD-II* gives the opportunity to solve differential algebraic equations (DAE) in the form

$$\begin{aligned}\mathbf{B}(t,\mathbf{x}(t),\mathbf{z}(t),\mathbf{u}(t),\boldsymbol{\eta})\ \frac{d}{dt}\mathbf{x}(t) &= \mathbf{f}(t,\mathbf{x}(t),\mathbf{z}(t),\mathbf{u}(t),\boldsymbol{\eta})\\ 0 &= \mathbf{g}(t,\mathbf{x}(t),\mathbf{z}(t),\mathbf{u}(t),\boldsymbol{\eta}).\end{aligned} \tag{1}$$

The vectors $\mathbf{x}(t)$ and $\mathbf{z}(t)$ describe the differential and algebraic system states. The system behavior is controlled by the control vector $\mathbf{u}(t)$ and the global design parameter vector $\boldsymbol{\eta}$. The objective function is of generalized Bolza type, containing Lagrange and Mayer terms

$$J = \Phi(T,\mathbf{x}(T),\mathbf{z}(T),\boldsymbol{\eta}) + \int_{t_0}^{T} L(t,\mathbf{x}(t),\mathbf{z}(t),\boldsymbol{\eta})dt. \tag{2}$$

For solving the original continuous problem it is reformulated as a nonlinear programming problem, approximating the control functions by a piecewise representation on a finite number of subintervals, called multiple shooting intervals. In the direct multiple shooting method, the DAE is solved independently on each subinterval. This is done by an iterative solution procedure, a specially tailored sequential quadratic programming algorithm. A far more complete description of the methods employed is given by Leineweber [15]. The optimization formulation to determine the PTPTO motion subject to physical inequality constraints $\mathbf{h}(\mathbf{x},\mathbf{u}) \in \mathbb{R}^m$ as well as a general dynamical system representing equations of motion in state-space representation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u})$ is given by

$$\min_{T,\mathbf{u}(\cdot)} J = \int_0^T 1dt \tag{3a}$$

$$\text{s.t.} \qquad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u}) \tag{3b}$$

$$\mathbf{h}(\mathbf{x},\mathbf{u}) \leq \mathbf{h}_b \tag{3c}$$

$$\mathbf{x}(0) = \mathbf{x}_0\ ,\ \mathbf{x}(T) = \mathbf{x}_{end} \tag{3d}$$

$$\text{for} \quad t \in [0,T]. \tag{3e}$$

The dependency on time is suppressed for simplicity from now on. As a result the optimal system input $\mathbf{u}^*$ as well as the state trajectory $\mathbf{x}^*$ are obtained.

III. Dynamic Movement Primitives

At first an introduction to DMPs is given and in the following the approach for utilizing these in the field of time-optimal trajectory planning is explained.

A. Introduction

Dynamic movement primitives have been introduced first to humanoid robotics as a new approach to movement planning ([13], [16]). The DMP as a new class of movement generators has its origin in the field of neurophysiology, due to the motivation from convergent force fields that are observed in frogs after spinal-cord stimulation [17]. The most advantageous property of a DMP, in particular for this contribution, is the simple adaption of a given path to a new target position. Using DMPs, a trajectory q is represented by a set of differential equations, that describe a spring-damper system with a nonlinear disturbance f and a goal position g

$$\begin{aligned}\tau\dot{q} &= v\\ \tau\dot{v} &= K(g-q) - Dv + (g-q_0)f(s),\end{aligned} \tag{4}$$

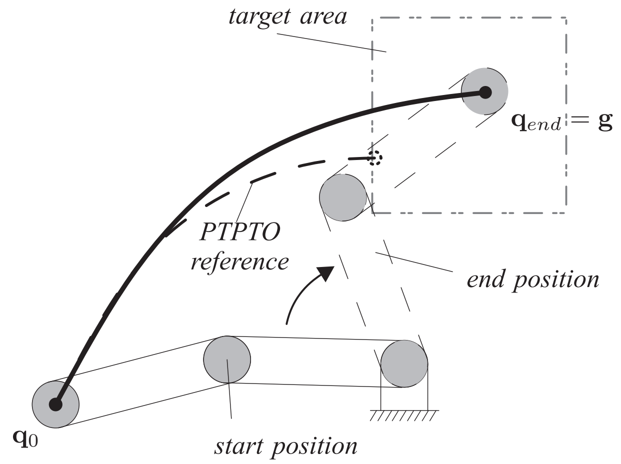

Fig. 1. Advantages of Dynamic Movement Primitives; Bold line: DMP path / Dashed line: PTPTO reference path

also known as *acceleration DMP*. The nonlinear function f, depending on a phase variable s

$$f(s) = \frac{\sum_i \Psi_i(s) w_i}{\sum_i \Psi_i(s)} s \tag{5}$$

describes normalized sums of Gaussian functions

$$\Psi_i(s) = \mathrm{e}^{-h_i(s-c_i)^2} \tag{6}$$

weighted with variable parameters $\mathbf{w} = [w_1, \ldots, w_k]$. Therewith a desired trajectory q_d can be reproduced. The variables c_i and h_i denote the location and shape of the ith function. The phase variable s, going from 1 to 0 is used for avoiding explicit time dependency, described by an additional canonical system

$$\tau\dot{s} = -\alpha s. \tag{7}$$

K and D are chosen such that critical damping with a double-fold eigenvalue of the system in (4) at $\lambda_{1,2} = D/2$ and $K = D^2/4$ is attained. Using this, $\alpha = D/3$ is a suitable constant for the canonical system (7). The benefit using this transcription to make the nonlinear function f not depending on time directly is, that movement duration can be changed by adjusting τ only due to the scale-invariance of the differential equations. As introduced before, trajectories for a manipulator used for pick-and-place or similar tasks deviate only in a very small range from each other, because of varying target positions. Hence the idea is to utilize the offline generated PTPTO motion from II as reference for an adaption in a suitable manner. Furthermore the property of DMPs concerning the variation of the goal position is advantageous. Thereby the whole movement is adapted as a result of the unique point attractor, see (4). For a simple two-dimensional planar system, this is sketched in Fig. 1.

B. Trajectory learning

For utilizing a PTPTO reference trajectory to generate a nearly time-optimal movement primitive, the required procedure of imitation learning (learning from a trajectory) is briefly explained for an one-dimensional system. Assuming a desired

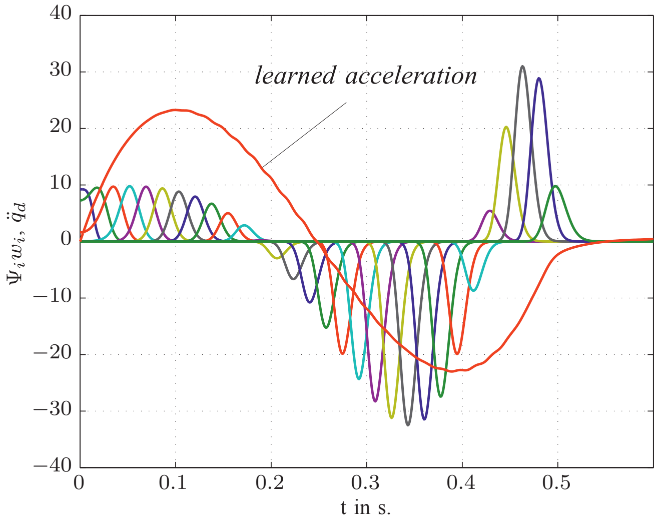

Fig. 2. Weighted Gaussian functions $\Psi_i w_i$ and learned acceleration $\ddot{q}_d$

trajectory q_d, $\dot{q}_d$ and $\ddot{q}_d$, (4) is reformulated to obtain the nonlinear function

$$f = \frac{\tau_d^2 \ddot{q}_d + D\tau_d \dot{q}_d - K(g_d - q_d)}{(g_d - q_{d,0})}. \tag{8}$$

Then (7) is integrated to obtain the phase variable $s(t)$. Based on this information the weighting parameters $\mathbf{w}$ can be calculated using different strategies ([10]), e.g. with locally weighted linear regression or locally weighted projection regression techniques, what is used here. Thereby the location c_i and the shape h_i are learned simultaneously. A recent review of learning the weighting parameters is given in [11].

In Fig. 2 the weighted Gaussian functions are shown in comparison to the acceleration $\ddot{q}_d$ of a desired (learned) sample trajectory.

For multi-dimensional systems, each degree-of-freedom $r \in \mathbb{N}$ has to be described by (4) with its own set of parameters $\mathbf{w}_r$. Nevertheless one single canonical system (7) with a single phase variable s can be used for all dimensions. Utilizing these methods, the time-optimal trajectory $\mathbf{x}^* = \left[\mathbf{q}_d^T(\mathbf{g}_d, \tau_d) \quad \dot{\mathbf{q}}_d^T(\mathbf{g}_d, \tau_d)\right]^T$ with $\mathbf{q}_d = \left[q_{d,1}(g_{d,1}, \tau_d) \ldots q_{d,r}(g_{d,r}, \tau_d)\right]^T$ from II is learned with DMPs to obtain a set of optimal parameters $\mathbf{w}_r^*$ for the imitated trajectories $\mathbf{x}^*_{dmp}$. Based on this time-optimal movement primitive, new state trajectories $\mathbf{x}_{dmp}(\mathbf{g}, \tau)$ are generated by adjusting the goal positions $\mathbf{g} \in \mathbb{R}^r$ for all DoF and the movement duration τ.

C. Time-scaling estimation

As a result of the optimization problem, the given constraints are violated in general, when moving on the trajectories $\mathbf{x}_{dmp}(\mathbf{g}, \tau)$ with adjusted goal positions $\mathbf{g}$. Therefore a strategy is developed in order to reconsider the violated bounds. A very straightforward way is to scale the trajectories in time in a convenient way to obtain nearly time-optimal trajectories $\mathbf{x}^{*,sub}_{dmp}$. The most challenging part is to provide a sufficient estimation for the required time-scaling factor. Doing a first order Taylor series expansion of the inequality constraints of the optimization problem in (3c) w.r.t. new goal positions $\mathbf{g}$ close to $\mathbf{g}_d$ and the movement duration τ, the following equation must hold for respecting the inequality constraints

$$\underbrace{\mathbf{h}(\mathbf{x}^*_{dmp}, \mathbf{u}^*)}_{\mathbf{h}(t)} + \underbrace{\left.\frac{\partial \mathbf{h}}{\partial \mathbf{g}}\right|_{(\mathbf{g}=\mathbf{g}_d, \tau=\tau_d)} \Delta \mathbf{g}}_{\mathbf{h}_{\mathbf{g}}(t)} + \underbrace{\frac{\partial \mathbf{h}}{\partial \tau}}_{\nabla_\tau \mathbf{h}(t)} \Big|_{(\mathbf{g}=\mathbf{g}_d, \tau=\tau_d)} \Delta\tau \leq \mathbf{h}_b. \tag{9}$$

Thereby $\Delta\tau$ denotes a scaling in time of the DMP trajectories for each joint. In order to get an estimation of an appropriate value for this variable, the following optimization problem is solved

$$\begin{aligned} \max_{i,t} \Delta\tau &= \left\{ \frac{h_i(t) + h_{\mathbf{g},i}(t) - h_{b,i}}{-\nabla_\tau h_i(t)} \right\} \\ \text{s.t.} \qquad & h_i(t) + h_{\mathbf{g},i}(t) > h_{b,i} \end{aligned} \tag{10}$$

with $i \in [1, \ldots, m]$ denoting the ith entry of the specified vector. Loosely speaking, time is scaled such that the constraint violation, that causes maximum necessary time scaling, is compensated. Due to nonlinearity of the constraints in (3c) and the consequential linearization error, it may be necessary to increase the time-scaling factor to guarantee conservative boundedness. Based on that, the r nearly time-optimal trajectories can be calculated using DMPs

$$\begin{aligned} \tau \dot{q}_r &= v_r \\ \tau \dot{v}_r &= K(g_r - q_r) - Dv_r + (g_r - q_{r,0}) f(s) \end{aligned} \tag{11}$$

with $\tau = \tau_d + \Delta\tau$ for all r DoFs in order to stay on the original geometric path.

D. Computation analysis

This procedure which generates such nearly time-optimal point-to-point trajectories is divided into *offline* (a priori) computable parts

(i) Computation of a PTPTO reference motion
(ii) Learning process of these trajectories using DMPs
(iii) Generation of the variables $\mathbf{h}, \partial\mathbf{h}/\partial\mathbf{g}, \nabla_\tau \mathbf{h}$
(iv) Estimation of scaling factor $\Delta\tau$ using (10)

and an *online* computable one

(I) Generation of new joint trajectories $\mathbf{x}^{*,sub}_{dmp}$ using (11).

Steps (i)-(iii) are done just once and only (iv) has to be computed for every new goal position in the first sampling step. Although (iv) has to be done offline, it is limited to approximately

$$n_{ops} \approx mN_s\,(2r+4) = O(N_s) \tag{12}$$

arithmetic and comparison operations and can be computed in real-time. The number of sampling steps $N_s = T/T_s$ is due to time discretization with sampling time T_s. m denotes the number of inequality constraints, see (3c). Part (I) results in just a simple integration for each joint and is processed in every sampling step.

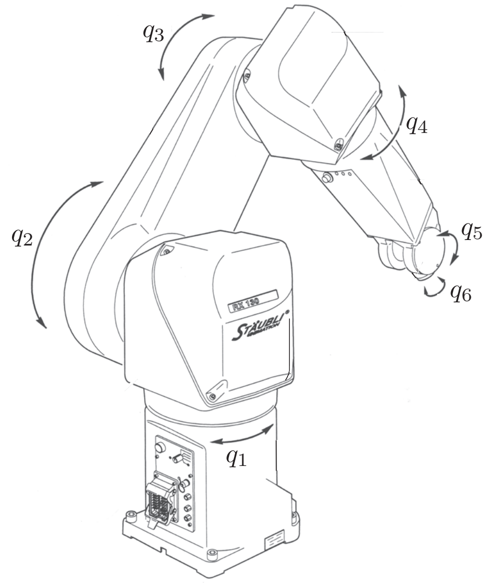

Fig. 3. Stäubli RX130L industrial robot

IV. Application example

The robot under consideration, shown in Fig. 3, is a Stäubli RX130L industrial robot with a maximum payload of $10\,\mathrm{kg}$ and $r = 6$ DoF.

A. Modeling

The mathematical model of the considered system with N bodies is obtained with the Projection Equation

$$\sum_{i=1}^{N} \begin{bmatrix} \left(\frac{\partial_R \mathbf{v}_C}{\partial \dot{\mathbf{q}}}\right)^T \\ \left(\frac{\partial_R \boldsymbol{\omega}_C}{\partial \dot{\mathbf{q}}}\right)^T \end{bmatrix}_i^T \begin{bmatrix} \left({}_R\dot{\mathbf{p}} + {}_R\tilde{\boldsymbol{\omega}}_{IR\,R}\mathbf{p} - {}_R\mathbf{f}^e\right) \\ \left({}_R\dot{\mathbf{L}} + {}_R\tilde{\boldsymbol{\omega}}_{IR\,R}\mathbf{L} - {}_R\mathbf{M}^e\right) \end{bmatrix}_i = \mathbf{0}, \tag{13}$$

see [18]. The $\tilde{(\ )}$ operator denotes the skew-symmetric matrix representing the cross product ($\tilde{\mathbf{a}}\mathbf{b} = \mathbf{a} \times \mathbf{b}$). This method projects the linear $\mathbf{p}$ and angular momenta $\mathbf{L}$, formulated in an arbitrary reference frame R, with the Jacobian matrices $\left(\partial_R \mathbf{v}_C / \partial \dot{\mathbf{q}}\right)^T$ and $\left(\partial_R \boldsymbol{\omega}_C / \partial \dot{\mathbf{q}}\right)^T$ into the direction of unconstrained motion. In this formulation ${}_R\mathbf{v}_C$ and ${}_R\boldsymbol{\omega}_C$ denote the linear and angular velocities of the center of gravity of each body i. The variables ${}_R\mathbf{f}^e$ and ${}_R\mathbf{M}^e$ denote impressed forces and moments. Based on that, the equations of motion in minimal description are obtained

$$\begin{aligned} \mathbf{M}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}) &= \mathbf{Q} = \mathbf{B}\mathbf{u} \\ \mathbf{u} &= \mathbf{M}_{mot} \end{aligned} \tag{14}$$

with minimal coordinates $\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 \end{bmatrix}^T$. Terms due to gravity, friction and all other nonlinear effects like Coriolis and centrifugal forces can be found in $\mathbf{g}(\mathbf{q}, \dot{\mathbf{q}})$. $\mathbf{M}$ represents the mass matrix and $\mathbf{B}$ contains the gear ratios and maps the actuator torques $\mathbf{M}_{mot}$ into minimal space.

For this system, the optimization problem in (3) is written using the state vector $\mathbf{x} = \begin{bmatrix} \mathbf{q}^T & \dot{\mathbf{q}}^T \end{bmatrix}^T$ as well as the inequality constraints $\mathbf{h} = \begin{bmatrix} \mathbf{h}_1^T & \mathbf{h}_2^T \end{bmatrix}^T$ with

$$\begin{aligned} h_{1,i} &= |\dot{q}_i(\mathbf{x}, \mathbf{u})| \leq \dot{q}_{b,i} \quad , \quad i \in [1 \ldots 6] \\ h_{2,i} &= |u_i| \leq Q_{b,i} \end{aligned} \tag{15}$$

for bounds on angular rates $\dot{q}_{b,i}$ as well as joint torques $Q_{b,i}$. The time-optimal solution $\mathbf{x}^*$ of (3) in combination with the inequality constraints in (15) is obtained with *MUSCOD-II* . This is used for generating a set of optimal parameters $\mathbf{w}_r^*$ to obtain a nearly time-optimal movement primitive for each joint. With the calculation of the vectors $\mathbf{h}, \partial\mathbf{h}/\partial\mathbf{g}, \nabla_\tau \mathbf{h}$ the last step of the offline computation, that is required once, is done.

V. Results

This section shows optimization results and draws a comparison between offline generated time-optimal *MUSCOD-II* solutions and the nearly time-optimal DMP solution approach. The start and end points used for PTPTO motion optimization are obtained using inverse kinematics $\mathbf{q}_0(\mathbf{z}_0)$ and $\mathbf{q}_{end}(\mathbf{z}_{end})$ with

$$\mathbf{z}_0 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} = \begin{pmatrix} 1.26 \\ -0.4 \\ 0.33 \end{pmatrix} \mathrm{m}, \mathbf{z}_{end} = \begin{pmatrix} x_{end} \\ y_{end} \\ z_{end} \end{pmatrix} = \begin{pmatrix} 0.36 \\ 0.5 \\ 0.73 \end{pmatrix} \mathrm{m} \tag{16}$$

and a constant orientation. Therewith the equality constraints $\mathbf{x}_0 = \begin{bmatrix} \mathbf{q}_0^T & \mathbf{0} \end{bmatrix}^T$ and $\mathbf{x}_{end} = \begin{bmatrix} \mathbf{q}_{end}^T & \mathbf{0} \end{bmatrix}^T$ in (3d) are given. The bounds are $\dot{\mathbf{q}}_b = [4.45, 4.45, 5.7, 6.55, 7.68, 18.0]^T$ rad/s and $\mathbf{Q}_b = [9.05, 9.05, 6.4, 4.07, 6.27, 4.43]^T$ Nm. The time-optimal trajectory $\mathbf{x}^*$ takes a minimal time of $T = 0.6744\,\mathrm{s}$ and is imitated by DMPs using $k = 300$ Gaussian functions for each joint. A high number of Gaussian basis functions is necessary in order to represent a time-optimal trajectory sufficiently. The normalized joint torques resulting from the PTPTO reference motion, denoted with a "*", can be seen in Fig. 4, hence only the relevant ones of axes $1, 2$ and 3 are shown in further plots. For this reference motion, the derivatives $\dot{\mathbf{Q}}$ are slightly constrained as well to obtain smooth actuator torques.

A. Comparison to MUSCOD-II solution

Since this method is suitable for small deviations from the reference, the goal positions g_i , $i \in [1..6]$ of the time-optimal movement primitives are varied such that the resulting TCP position is in a symmetric cube with equal edge length of $10\,\mathrm{cm}$, see Fig. 5. This is equivalent to a mean deviation of $5.47\,\%$ of the nominal path movement $\mathbf{q}_{end} - \mathbf{q}_0$. Figure 6 and 7 show the joint angles as well as the normalized joint torques resulting from one of these time-optimal DMP trajectories with a mean variation of about $5\,\%$ of the goal positions in comparison to the PTPTO reference motion. The estimated time-scaling factor $\Delta\tau$ scales to a new minimal time $T_{dmp} = 0.72\,\mathrm{s}$, that results in an optimality of $Opt. = T/T_{dmp} = 94.4\,\%$ with the time-optimal solution $T \approx 0.68\,\mathrm{s}$. Table I shows an optimality analysis for further different nearly time-optimal trajectories in comparison to the respective *MUSCOD-II* solution to generalize optimality for this specific example. For such small variations, the time-optimal solution T does not vary noticably w.r.t. minimal

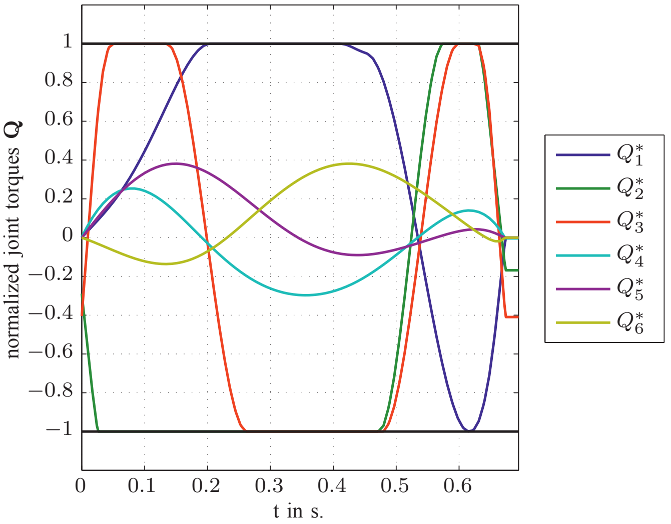

Fig. 4. Normalized joint torques for the PTPTO reference motion $\mathbf{x}^*$ obtained with *MUSCOD-II*

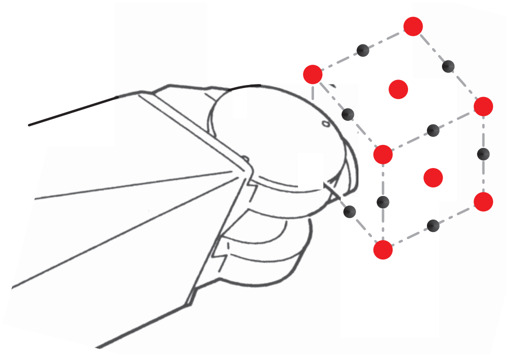

Fig. 5. Cubic range around the reference target position

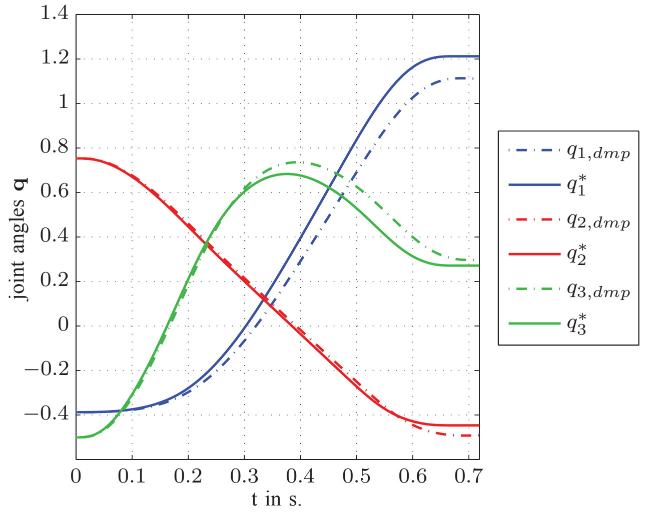

Fig. 6. Joint angles of axes $1, 2, 3$ for a specific nearly time-optimal point-to-point dynamic movement primitive

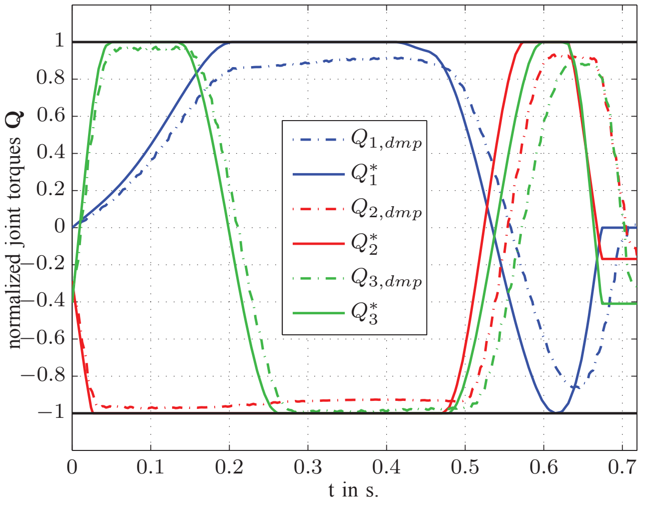

Fig. 7. Relevant (axes $1, 2, 3$) normalized joint torques for a specific nearly time-optimal point-to-point dynamic movement primitive

time, hence $T \approx 0.68\,\mathrm{s}$ for all PTPTO motions obtained with *MUSCOD-II* . The variable $\|\mathbf{d}_{TCP}\|_2$ denotes the euclidean distance between the reference target (Ref.) and the new goal position of the TCP. As one can see, the durations T_{dmp} of the resulting nearly time-optimal point-to-point trajectories are close to optimality with an average of 95 % in this cubic range.

TABLE I. OPTIMALITY ANALYSIS FOR DIFFERENT TRAJECTORIES

Traj.	$\|\mathbf{d}_{TCP}\|_2$ [m]	T_{dmp} [s]	T [s]	*Opt.* [%]
1	0.05	0.71	≈0.68	95.5
2	0.07	0.71	≈0.68	95.4
3	0.05	0.71	≈0.68	95.3
4	0.07	0.73	≈0.68	93.4
5	0.07	0.72	≈0.68	94.6
6	0.087	0.71	≈0.68	95.5
7	0.07	0.73	≈0.68	93.8
8	0.05	0.70	≈0.68	97.0
9	0.07	0.72	≈0.68	94.7
10	0.087	0.71	≈0.68	95.3

B. Comparison to PCTOM

For verification of the resulting time-optimality of the proposed method, a comparison to state-of-the-art time-optimal trajectory planning methods on predefined geometric paths is drawn. Due to their real-time capability, they are already used and relevant in industry nowadays. Unfortunately it is very hard to predefine a geometric path that yields a similar behavior like the PTPTO motion. As simplest path for the PCTOM a straight line in space between the start and end points $\mathbf{z}_0$ and $\mathbf{z}_{end}$ is calculated exemplarily. Further difficulties like singularities in inverse kinematics can occur, what is the case here. As expected, the resulting minimum time $T = 1.048\,\mathrm{s}$ is much higher than with the geometric point-to-point path. This is also due to the fact, that beneath the joint torques, the angular velocity constraints are going active near the singularity, shown in Fig. 8 and 9. A detailed explanation

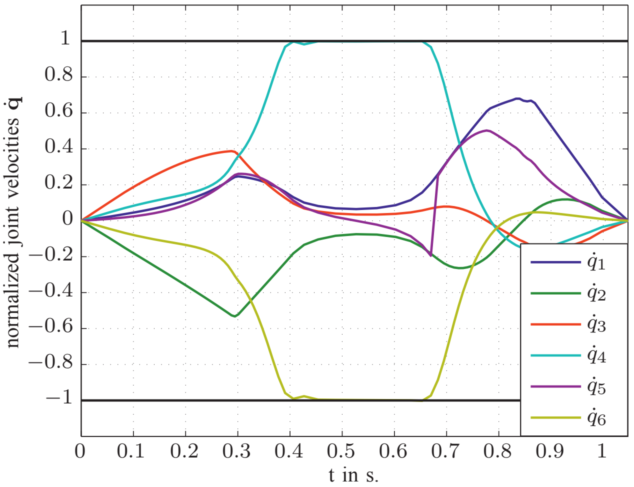

Fig. 8. Joint velocities for a path-constrained time-optimal motion on a straight line between $\mathbf{z}_0$ and $\mathbf{z}_{end}$ near a singularity

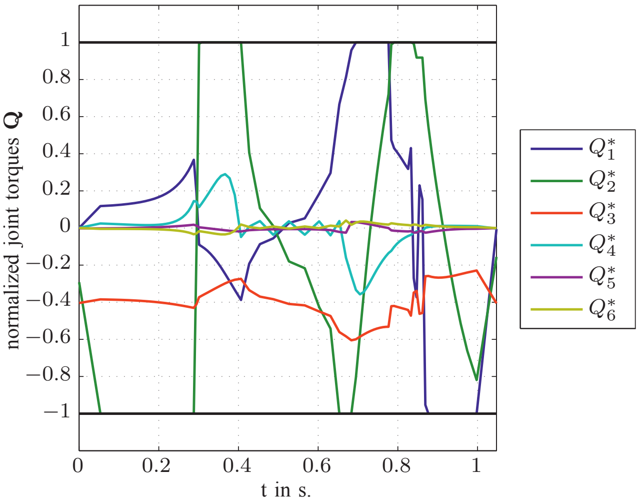

Fig. 9. Joint torques for a path-constrained time-optimal motion on a straight line between $\mathbf{z}_0$ and $\mathbf{z}_{end}$ near a singularity

of time-optimal trajectory planning for point-to-point as well as path constrained motions can be found in [19].

VI. Conclusion

In this paper a real-time capable concept for nearly time-optimal point-to-point trajectory-planning using DMPs is introduced. Therefore a PTPTO reference trajectory is generated and used as a basis for the offline calculation of a time-optimal movement primitive. To overcome the problem of constraint violation when varying the TCP goal position of the DMP online, a time-scaling estimation approach is proposed to generate nearly time-optimal motion trajectories. A computation analysis demonstrates the real-time capability of the algorithm. In a last step the concept is applied to a six DoF serial robot and constraint consideration as well as an optimality analysis in comparison to an offline generated solution are shown.
Due to the fact that the current strategy only allows a variation in a small range around the goal position, future work will focus on extending this to multiple regions.

Acknowledgment

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] F. Pfeiffer and R. Johanni, “A concept for manipulator trajectory planning,” *IEEE Journal of Robotics and Automation*, vol. 3, no. 2, pp. 115 –123, 1987.

[2] D. Constantinescu and E. A. Croft, “Smooth and time-optimal trajectory planning for industrial manipulators along specified paths,” *Journal of Robotic Systems*, vol. 17, no. 5, pp. 233–249, 2000.

[3] D. Verscheure, B. Demeulenaere, J. Swevers, J. De Schutter, and M. Diehl, “Time-energy optimal path tracking for robots: a numerically efficient optimization approach,” in *10th IEEE International Workshop on Advanced Motion Control*, 2008, pp. 727 –732.

[4] J. T. Betts and W. P. Huffman, “Path-constrained trajectory optimization using sparse sequential quadratic programming,” *Journal of Guidance, Control, and Dynamics*, vol. 16, no. 1, pp. 59–68, 1993.

[5] L. V. den Broeck, M. Diehl, and J. Swevers, “A model predictive control approach for time optimal point-to-point motion control,” *Mechatronics*, vol. 21, no. 7, pp. 1203 – 1212, 2011.

[6] L. Consolini and A. Piazzi, “Generalized bang-bang control for feed-forward constrained regulation,” *Automatica*, vol. 45, no. 10, pp. 2234 – 2243, 2009.

[7] P. Janssens, G. Pipeleers, and J. Swevers, “Model-free iterative learning of time-optimal point-to-point motions for LTI systems,” in *50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, 2011, pp. 6031–6036.

[8] D. Verscheure, M. Diehl, J. De Schutter, and J. Swevers, “Recursive log-barrier method for on-line time-optimal robot path tracking,” in *American Control Conference, 2009. ACC ’09.*, 2009, pp. 4134 –4140.

[9] L. Messner, P. Staufer, H. Gattringer, and H. Bremer, “A time optimal trajectory generation method for a flatness based controlled flexible link robot,” in *Proceedings of the 5th European Conference on Structural Control*, 2012, p. 8.

[10] C. G. Atkeson, A. W. Moore, and S. Schaal, “Locally weighted learning,” *Artificial Intelligent Review*, vol. 11, no. 1-5, pp. 11–73, 1997.

[11] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, “Dynamical movement primitives: Learning attractor models for motor behaviors,” *Neural Computation*, vol. 25, no. 2, pp. 328–373, 2013.

[12] A. Ude, A. Gams, T. Asfour, and J. Morimoto, “Task-specific generalization of discrete and periodic dynamic movement primitives,” *Trans. Rob.*, vol. 26, no. 5, pp. 800–815, 2010.

[13] A. Ijspeert, J. Nakanishi, and S. Schaal, “Movement imitation with nonlinear dynamical systems in humanoid robots,” in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, 2002, pp. 1398–1403.

[14] M. Diehl, D. B. Leineweber, and A. Schäfer, *MUSCOD-II Users’ Manual*. Universität Heidelberg: IWR 2001-25, 2001.

[15] D. Leineweber, “Efficient reduced SQP methods for the optimization of chemical processes described by large sparse DAE models,” *Fortschritt-Berichte VDI Reihe 3*, vol. 613, 1999.

[16] A. J. Ijspeert, J. Nakanishi, and S. Schaal, “Learning attractor landscapes for learning motor primitives,” in *Advances in Neural Information Processing Systems 15*. MIT Press, 2002, pp. 1547–1554.

[17] S. Giszter, F. Mussa-Ivaldi, and E. Bizzi, “Convergent force fields organized in the frogs spinal cord,” *Journal of Neuroscience*, vol. 13, pp. 467 – 491, 1993.

[18] H. Bremer, *Elastic Multibody Dynamics: A Direct Ritz Approach*. Springer-Verlag, 2008.

[19] H. Gattringer, R. Riepl, and M. Neubauer, “Optimizing industrial robots for accurate high-speed applications,” *Journal of Industrial Engineering*, vol. 2013, p. 12.