2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Computationally Efficient Trajectory Optimization for Linear Control Systems with Input and State Constraints

Jean-François Stumper and Ralph Kennel, *Senior Member, IEEE*

Abstract— **This paper presents a trajectory generation method that optimizes a quadratic cost functional with respect to linear system dynamics and to linear input and state constraints. The method is based on continuous-time flatness-based trajectory generation, and the outputs are parameterized using a polynomial basis. A method to parameterize the constraints is introduced using a result on polynomial nonpositivity. The resulting parameterized problem remains linear-quadratic and can be solved using quadratic programming. The problem can be further simplified to a linear programming problem by linearization around the unconstrained optimum. The method promises to be computationally efficient for constrained systems with a high optimization horizon. As application, a predictive torque controller for a permanent magnet synchronous motor which is based on real-time optimization is presented.**

I. Introduction

Trajectory optimization in real-time is an important part of many modern control systems, for instance, predictive control. The trajectory optimization problem must be solved in the sampling interval, determined by the plant dynamics. A computationally efficient solution, however, encounters two major obstacles: first, the optimization horizon has a minimum length, either to avoid suboptimality or as stability criterion, and second, the trajectory must satisfy input and state constraints to be feasible. For fast systems, many of the existing constrained predictive control schemes, which are based on numerical iterations, are not applicable.

Concerning the horizon length problem, the continuous approach to predictive control is of interest [1]. Using differential flatness, the optimal control problem can be rewritten as output optimization problem. The basis function approach is applied to obtain a finite-parameter optimization problem [2] [3] [4] [5], analogeously to discrete-time optimization. A long optimization horizon is, however, obtained with comparably few optimization parameters.

A remaining issue regarding computational efficiency is the inclusion of constraints. The classical method is the application of penalty functions [3]. As an alternative, a coordinate transformation was proposed to modify the constrained problem to a nonlinear unconstrained problem [6], solvable with nonlinear calculus of variations methods. The existing optimization methods with penalty functions or nonlinear coordinate transformations lead to a nonlinear or non-linear-quadratic problem, increasing the computational burden. If only feasibility is of interest, numerical procedures applying time scaling to slow down some variables can be used [1].

This paper treats the special case of the linearly constrained linear-quadratic problem. The linear structure of the system is exploited in the transformation of the problem to a finite-parameter optimization problem. The linear-quadratic structure of the problem is maintained. The unconstrained problem is solved algebraically as it is convex. A quadratic or linear programming (QP/LP) solver is then applied for constraint handling to modify the solution to a feasible trajectory. The solution is computationally efficient, as first a continuous parameterization is performed, and second a linear programming solver is used for the quadratic problem by linearizing around the unconstrained optimum. To the knowledge of the authors, QP and LP have so far not been applied to this type of continuous problems.

II. Preliminaries: Continuous Flatness-Based Linear-Quadratic Trajectory Optimization

A. Optimal Control Problem

The optimal control problem considered is to generate trajectories on a finite time horizon T minimizing the quadratic cost functional

$$J = \int_0^T \left(\mathbf{x}^{\mathrm{T}}(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^{\mathrm{T}}(t)\mathbf{R}\mathbf{u}(t)\right)\ \mathrm{dt} + (\mathbf{x}(T) - \mathbf{x}^*)^{\mathrm{T}}\mathbf{P}(\mathbf{x}(T) - \mathbf{x}^*) \tag{1}$$

with states $\mathbf{x}(t) \in \mathbb{R}^n$, inputs $\mathbf{u}(t) \in \mathbb{R}^m$, weight matrices $\mathbf{P} \in \mathbb{R}^{n\times n}$, $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{m\times m}$, and the desired final state $\mathbf{x}^* \in \mathbb{R}^n$. The weight matrices are assumed to be positive definite and symmetric. Any quadratic functional is eligible, for instance the reference $\mathbf{x}^*$ can also be included in the cost integral. Quadratic cost functionals are preferred in predictive control as they will provide good closed-loop behavior [7]. They can also describe physical costs better than other norms. Furthermore, if the weight matrices are positive definite, the optimization problem is convex and easily solvable: a unique optimum exists and is found by solving first-order necessary conditions. The states and inputs are constrained to the time invariant linear dynamics of the multi-input multi-output system

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{2}$$

with system matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ and input matrix $\mathbf{B} \in \mathbb{R}^{n\times m}$. The system is further assumed to be controllable. The optimization is subject to N_c linear input and state constraints

$$\mathbf{G}_x\mathbf{x}(t) + \mathbf{G}_u\mathbf{u}(t) + \mathbf{g}_0 \le 0,\ \ \forall t \in [0, T] \tag{3}$$

with $\mathbf{G}_x \in \mathbb{R}^{N_c\times n}$, $\mathbf{G}_u \in \mathbb{R}^{N_c\times m}$, $\mathbf{g}_0 \in \mathbb{R}^{N_c}$, as well as the n initial conditions

$$\mathbf{x}(0) = \mathbf{x}_0. \tag{4}$$

Terminal constraints could be imposed too, but may yield an unfeasible problem in the presence of input or state bounds.

B. Parameterization of the System Variables using Flatness

By definition, a linear system is said to be differentially flat if there exist m output functions

$$\mathbf{y}_f(t) = \mathbf{C}_f\mathbf{x}(t), \tag{5}$$

with $\mathbf{y}_f(t) \in \mathbb{R}^m$, $\mathbf{C}_f \in \mathbb{R}^{m\times n}$, such that all states $\mathbf{x}(t)$ and inputs $\mathbf{u}(t)$ can be expressed as linear combination of the flat outputs and a finite number of their derivatives [8] [9]. Differential flatness can be interpreted as transformation into controller canonical form, and in the linear case, it is equivalent to controllability [9]. The flat outputs $\mathbf{y}_f(t)$ are

This work was supported through the National Research Funds of Luxembourg under grant PhD-08-070.

J-F. Stumper and R. Kennel are with the Institute of Electrical Drive Systems and Power Electronics, Department of Electrical Engineering and Information Technology, Technische Universität München, Arcisstr. 21, D-80333 Munich, Germany. jean-francois.stumper@tum.de

978-1-4577-0079-8/11/$26.00 ©2011 AACC

thus the controller canonical form outputs, and the canonical form state vector is

$$\mathbf{z}(t) = \left(\left(y_{f1}, \dot{y}_{f1}, .., y_{f1}^{(r_1-1)}\right), \left(y_{f2}, \dot{y}_{f2}, .., y_{f2}^{(r_2-1)}\right), \\ .., \left(y_{fm}, \dot{y}_{fm}, .., y_{fm}^{(r_m-1)}\right)\right)^{\mathrm{T}} \quad (6)$$

with $\mathbf{z}(t) \in \mathbb{R}^n$ and where r_i is the corresponding vector relative degree with regard to the output y_{fi} (y_{fi} being the i-th flat output resp. i-th element of $\mathbf{y}_f$ in (5), see [9]). Note that $\sum_{i=1}^{m} r_i = n$. The differential parameterization of the system variables is [8] [9]

$$\mathbf{x} = \Xi_x \left(\mathbf{y}_f, \dot{\mathbf{y}}_f, .., \mathbf{y}_f^{(k-1)}\right), \quad (7)$$

$$\mathbf{u} = \Xi_u \left(\mathbf{y}_f, \dot{\mathbf{y}}_f, .., \mathbf{y}_f^{(k)}\right), \quad (8)$$

where $k \in \mathbb{R}$ is $k = \max\{r_i\}$. The functions Ξ_x and Ξ_u are linear functions of $\mathbf{z}$ resp. of $\mathbf{z}$ and $\dot{\mathbf{z}}$, as they are derived from the controller canonical form transformation. See that (7) is the inverse transformation from controller canonical form, and (8) follows from (7) and (2).

C. Output Parameterization with a Polynomial Basis

The Ritz-Galerkin method, also called basis function approach, is a direct method to find an approximate solution to an optimal control problem [10]. In the flatness-based approach, the outputs are parameterized as a linear combination of time-variant basis functions. It is comparable to the control parameterization Ritz method (CPRM) [11], where the inputs **u** are parameterized. Convergence of this method (combined with linear splines) was studied in [12] and more generally in [13].

As basis functions, polynomials are chosen. Their simplicity is exploited for the further developments, main advantage being that the parameters enter linearly, for instance they allow to prove that the transformed cost remains convex. They also allow constraint transformation based on a result on polynomial nonpositivity in section III. Furthermore, the dynamic shape of the resulting trajectory can be well analyzed.

Higher-order polynomials (for instance Laguerre and Legendre) have been used in many applications, but here, power series, the simplest form of polynomials, are used. The methods and results are all applicable to higher-order polynomials, there, the undetermined coefficients still enter linearly and they might inherit numerical advantages (power series are numerically stable only up to degree 12..15), but the notation would be less comprehensible. In [14] it was shown that only the polynomial degree but not the type of series are important for convergence. The choice is a simple approximate, but yet, it turns out to be quite accurate.

The system outputs are defined as degree N power series

$$y_{fi}(t) = \sum_{j=0}^{N} \alpha_{ij} \frac{t^j}{T^j} \quad , i = 1..m, \quad t \in [0, T] \quad (9)$$

with $\alpha_{ij} \in \mathbb{R}$. This is not an approximation of a system step response, but merely an assumption that the optimal solution can be described as a polynomial. The system response is exact, only the output is reduced in dimensionality.

The original problem is thereby transformed to a finite-parameter optimization problem where a set of constants is searched. Fig. 1 demonstrates the computational advantage. For instance, in the example in section IV, the prediction horizon is 2 ms (the slowest time constant in the system) and the sampling rate is 10 kHz (determined by the power stage). A discrete description takes 20 parameters, but here, only 6 are sufficient to describe a setpoint change. A more complicated trajectory may not be expected on most plants.

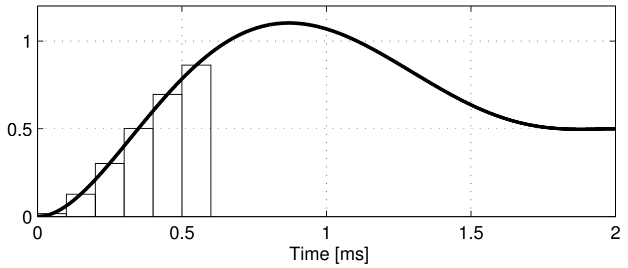

Fig. 1. Trajectories with 6 free parameters. Discrete-time horizon is maximum 0.6 ms, continuous trajectory is well-conditioned at the desired horizon length of 2 ms.

The n initial conditions (4) define n of the parameters α_{ij}, they are computed via the transformed state $\mathbf{z}(0)$ in (6) (See that the initial time is $t_0 = 0$, such that only the first element of the power series is nonzero, same for the derivatives). For further notational simplicity, the vector $\boldsymbol{\alpha}$ is defined as the vector of the undetermined coefficients

$$\boldsymbol{\alpha} = (\alpha_{1r_1}, .., \alpha_{1N}, \alpha_{2r_2}, .., \alpha_{2N}, .., \alpha_{mr_m}, .., \alpha_{mN})^{\mathrm{T}}, \quad (10)$$

with $\boldsymbol{\alpha} \in \mathbb{R}^{N'}$ with $N' = m \times N - n$, and the indices r_i represent the respective vector relative degree of the flat output y_{fi}.

Based on the flatness parameterization (7) and (8), and knowing that $\frac{d^i}{dt^i}\mathbf{y}_f$ are polynomials, the system variables can be parameterized with the coefficients $\boldsymbol{\alpha}$ yielding the functions

$$\mathbf{x}(t) = \boldsymbol{\Gamma}_x(\boldsymbol{\alpha}, t), \quad (11)$$

$$\mathbf{u}(t) = \boldsymbol{\Gamma}_u(\boldsymbol{\alpha}, t). \quad (12)$$

These functions are degree N polynomials of time t, where the coefficients $\boldsymbol{\alpha}$ enter linearly.

It is seen that $N \geq \max\{r_i\}$, $i = 1..m$, is mandatory to have degrees of freedom on the trajectory. It should be remarked that when choosing a higher N, the error bounds of the basis function approach become smaller and the dynamical response increases, at the cost of higher computational requirements and the risk of less good numerical conditioning of the parameters.

D. Remark: Generality of the Parameterization for Linear Systems with Power Series Outputs

It is remarked that the same parameterization (11) and (12) can be obtained for arbitrary non-flat outputs of a controllable and observable linear system, if the output is defined as polynomial series with undetermined coefficients [15]. Then, the outputs do not need to be redefined to flat, or controller canonical form outputs.

E. Parameterizing the Cost Functional

The parameterization of the system variables (11), (12) is applied to transform the cost functional (1) to a finite-parameter cost function.

Proposition II.1
Conditioning the cost functional.
The cost functional (1) with the substitutions (11) and (12) yields a biaffine function in the undetermined parameters $\boldsymbol{\alpha}$ *of the type*

$$J(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T \boldsymbol{K} \boldsymbol{\alpha} + \boldsymbol{k}^T \boldsymbol{\alpha} + k_0 \quad (13)$$

with $k_0 \in \mathbb{R}$, $\boldsymbol{k} \in \mathbb{R}^{N'}$ *and* $\boldsymbol{K} \in \mathbb{R}^{N' \times N'}$. ∎

The proof follows from a straightforward computation with the knowledge that (11), (12) are polynomials with affine coefficients $\boldsymbol{\alpha}$, and is omitted.

Eq. (13) is derived directly from (1) by substitution of (7), (8) and (9) (or (11) and (12)). Practical (symbolical) calculations are performed using a computer algebra tool, matrix $\mathbf{K}$ is the Hessian matrix of J, and vector $\mathbf{k}$ follows from the gradient of J.

Proposition II.2
Convexity of the conditioned cost function.
The conditioned cost function (13) is convex over the set $\boldsymbol{\alpha} \in \mathbb{R}^{N'}$ if matrices $\boldsymbol{Q}$, $\boldsymbol{R}$ and $\boldsymbol{P}$ in (1) are positive definite and real-symmetric matrices. ■

This statement of convexity is not obvious, as the transformed cost function (13) has a higher dimension as the original cost functional (1) (N' instead of n). It is proven for the unconstrained linear-quadratic case. This proof follows from the knowledge that the states and inputs are polynomials with coefficients linearly dependent on $\boldsymbol{\alpha}$, and is sketched in appendix I. As only linear constraints are regarded, the result can be assumed valid also for the constrained case.

This result allows to apply first-order conditions of optimality, and will guarantee a solution in finite time as an unique minimum of J exists.

III. Main Results: Constraint Handling

The previous section presented the unconstrained optimization of a controllable linear system with a flat output parameterized with a polynomial basis. This section introduces a computationally efficient method for including constraints in trajectory generation. It is a follow-up to the previous section.

A. Parameterizing the Constraints

The input and state constraints (3) are transformed with the system variable parameterizations (11) and (12). With the knowledge that the parameterizations (11) and (12) are univariate polynomials in t, each of the constraints in (3) is also a polynomial in t of order N where the coefficients are affine functions in $\boldsymbol{\alpha}$. Thus, the constraints are rewritten as

$$P_k(t) = \sum_{i=0}^{N} \left(g_{ki0} + \mathbf{g}_{ki}^T \ \boldsymbol{\alpha} \right) \ t^i \leq 0, \quad k = 1..N_c, t \in [0, T], \tag{14}$$

with $g_{ki0} \in \mathbb{R}$ and $\mathbf{g}_{ki} \in \mathbb{R}^{N'}$. Checking constraints over a time interval $t \in [0, T]$ is difficult. The typical solutions for such continuous problems are penalty functions in the cost functional, or nonlinear state transformations. In the following, the constraints are transformed to allow a direct check on $\boldsymbol{\alpha}$ independent of t, while maintaining the simplicity of a linear constraint.

Proposition III.1
Sufficient affine conditions for the constraints.
The univariate polynomials $P_k(t)$ are positive on the finite interval $t \in [0, T]$ if the conditions

$$P_k(0) \leq 0, \quad \forall k = 1..N_c, \tag{15}$$

$$P_k\left(p \frac{T}{N}\right) - \Delta \cdot P_k(0) \leq 0, \quad \forall p = 1..N, k = 1..N_c, \tag{16}$$

with a constant $\Delta \in \mathbb{R} > 0$ are satisfied. ■

The constraints, written in the introduced notations, are

$$\left(g_{k00} + \mathbf{g}_{k0}^T \ \boldsymbol{\alpha} \right) \leq 0, \quad k = 1..N_c \tag{17}$$

and

$$\sum_{i=0}^{N} \left(g_{ki0} + \mathbf{g}_{ki}^T \boldsymbol{\alpha} \right) \frac{T^i}{N^i} p^i - \Delta \cdot \left(g_{k00} + \mathbf{g}_{k0}^T \ \boldsymbol{\alpha} \right) \leq 0,$$
$$p = 1..N, k = 1..N_c. \tag{18}$$

The underlying idea is shown in Fig. 2. If a polynomial trajectory $P(t)$ of degree N is constrained at $N+1$ sampling points on the interval $[0, T]$ to be nonpositive, then, in the worst case, this polynomial reaches a maximum of $P(t) = -\Delta \cdot P(0)$. This maximum offset of $-\Delta \cdot P(0)$ is applied as interleaf to the constraint boundary (the second term in (16)). The proof of this proposition is presented in appendix II, where also the value of the constant Δ is found, which is a constant depending on the polynomial degree N only. After transformation of the constraints, they can be used directly in an optimization procedure, as the result is an affine and purely parametric constraint. They are affine functions of $\boldsymbol{\alpha}$.

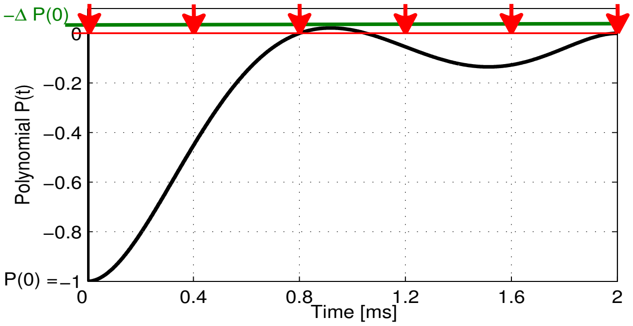

Fig. 2. A degree N polynomial trajectory (black) constrained at $N+1$ points to be nonpositive (red arrows) will not exceed the upper bound of $-\Delta \cdot P(0)$ (green line). Here, $N = 6$, $P(0) = -1$ and $\Delta = 0.037$.

The resulting set of constraints is sufficient, but not necessary, thus they guarantee maintainance of the constraints, but are too restrictive. The offset between the necessary and the sufficient conditions is the factor Δ, and the restriction is, for example, 6.4% for $N = 3$, 1.2% for $N = 10$, and even less for higher N. It is assumed that this restriction is acceptable for most applications regarding the inherited computational advantages.

Exact results on necessary and sufficient conditions on positivity of univariate polynomials over an interval $[0, T]$ exist. Semidefinite programming (SDP) is applied to establish linear matrix inequalities (LMI), which, just as for the presented results, are independent of t and can directly be used in a program [16]. These methods, however, require a numerical iterative algorithm to establish the LMIs, and can therefore not be considered if computational efficiency is important. Due to updated initial conditions, the constraints must be reconditioned in each trajectory generation cycle, and the computational burden adds up.

B. Trajectory Generation in Finite Time

In the previous section, the optimal control problem has been transformed to a finite-parameter optimization problem. The transformed problem can be solved in finite time using quadratic programming (QP) to find the parameters $\boldsymbol{\alpha}$, as the cost function is quadratic in $\boldsymbol{\alpha}$ and as the constraints are affine functions of $\boldsymbol{\alpha}$. This is an interesting result for continuous systems, as QP is a standard method in discrete-time optimization. Furthermore, the number of parameters is decoupled from the horizon length, thus higher prediction horizons can be reached with less parameters.

An even more convincing advantage of the presented method shall be a further reduction of the computational complexity. In the following, the problem will be approximated such that it is solvable using linear programming (LP) techniques. The approximation yields a feasible solution (thus it is satisfying the constraints) with a bounded suboptimality.

C. Transformation to a Least Distance Problem

In the first step, a transformation is performed to obtain a simpler cost functional.

Proposition III.2
Reformulation as least-distance problem.
The cost functional can be transformed to

$$J = \boldsymbol{f}^T\boldsymbol{f} + c, \tag{19}$$

where $\boldsymbol{f} = (f_1, f_2, .., f_{N'})^T \in \mathbb{R}^{N'}$ is the vector of transformed parameters and $c \in \mathbb{R}$ a constant. ■

The new parameter vector $\boldsymbol{f}$ is defined as

$$\boldsymbol{f} = \mathbf{F}(\boldsymbol{\alpha} - \boldsymbol{\alpha}_0), \tag{20}$$

where $\boldsymbol{\alpha}_0 \in \mathbb{R}^{N'}$ is the unconstrained minimum of the parameterized cost functional (13)

$$\boldsymbol{\alpha}_0 = -\frac{1}{2}\mathbf{K}^{-1}\mathbf{k}, \tag{21}$$

and matrix $\mathbf{F} \in \mathbb{R}^{N' \times N'}$ is defined such that

$$\mathbf{F}^{\mathrm{T}}\mathbf{F} = \mathbf{K} \tag{22}$$

with $\mathbf{K}$ from eq. (13). Computation is not an issue as $\mathbf{K}$ is positive definite and symmetric, for instance, in the notations in appendix I one would set $\mathbf{F} = \mathbf{BA}$. If this decomposition is not known, $\mathbf{F}$ can be computed via the Cholesky matrix decomposition $\mathbf{F}^{\mathrm{T}} = \mathrm{cholesky}(\mathbf{K}^{\mathrm{T}})$.

Transformation (20) must also be applied to transform the constraints. This is done by substituting the inverse transformation

$$\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 + \mathbf{F}^{-1}\boldsymbol{f} \tag{23}$$

to the constraint equation, yielding directly the affine constraints in terms of $\boldsymbol{f}$. The same equation is applied to retransform the obtained results into the original parameters. Contrary to the amount of parameters, the amount of constraints is not increased.

The least-distance problem is simpler to solve as the original QP problem, as some computations become obsolete in the iterations [17]. The transformation renders the quadratic programming more efficient and increases numerical stability.

D. Transformation to a Linear Programming Problem

Now the least distance problem can be approximated for a solution using linear programming. The L_2-norm is rewritten as L_1-norm. The variables in linear programs are limited to positive numbers, thus the variables $\boldsymbol{f}$ are replaced by

$$f_i = f_{ip} - f_{in}, \tag{24}$$

with $f_{ip}, f_{in} \in \mathbb{R}$ and $f_{ip} \geq 0$, $f_{in} \geq 0$.

Proposition III.3
Linear Cost Function for the Least-Distance Problem.
The cost function approximation

$$J = c + \sum_{i=1}^{N} f_i^2 \approx c + \sum_{i=1}^{N} |f_i| = c + \sum_{i=1}^{N} (f_{ip} + f_{in}) = J', \tag{25}$$

with $f_{ip}, f_{in} \geq 0$, yields a feasible solution with bounded suboptimality for a least-distance problem. ■

This approximation implies a large offset between the linearized cost J' and the correct cost J. However, this is not of importance, as the goal is to find a point in the parameter space $\boldsymbol{f}$ that is feasible and least-distance to the origin. Under this aspect, the suboptimality is not the variation of the cost, but of the difference of the found parameters in the squared distance to the origin. The contour lines of the quadratic cost are circles around the origin, whereas these of the linear cost are lozenges [7].

It should be remarked that $|f_i|$ is not necessarily equivalent to $f_{ip} + f_{in}$, but in the reverse, if $f_{ip} + f_{in}$ is minimized, it is equivalent to $|f_i|$ as $f_{ip}, f_{in} \geq 0$, such that at least one of each f_{ip}, f_{in} is zero.

The advantages of a linear programming solver are obvious, increased reliability and reduced computational complexity, and the computational burden only grows linearly with the degree of the output polynomial N. It represents a significant computational saving compared to when applying QP as solver.

If no constraint is active, the solution is exact. If a constraint is active, the worst case is when the active constraint vertex is parallel to a contour line (i.e. parallel to a side of the lozenge). It can be shown that the resulting cost is

$$J' = J_0 + N' \times J_C \tag{26}$$

in the worst case, where J' is the cost with the linear program, J_0 the cost of the unconstrained problem, J_C the extra cost when considering constraints in a quadratic program, and $N' = \dim(\boldsymbol{\alpha})$ the amount of free parameters $\boldsymbol{\alpha}$. Thus, the linear programming solver yields the worst-case suboptimality $J'/(J_0 + J_C)$ of up to N' times the cost. If not all constraints are active, or if the constraints intersect the contour lines, suboptimality will be considerably less. The bound of the suboptimality and the consideration that the overall suboptimality is relative to the cost of the unconstrained problem makes the simplification of a linear programming solver acceptable for many applications.

IV. Example: Predictive Torque Control of Permanent-Magnet Synchronous Machines

To present the advantages and the good functionality of the presented trajectory optimization scheme, a predictive torque controller for a permanent-magnet synchronous machine (PMSM) is presented. So far, this plant has been controlled via generalized predictive control (GPC), via finite control set MPC (FS-MPC) and via the explicit solution [18]. Even though the methods provided good results, they were either unconstrained, or limited to few prediction steps. The presented method obtains a high optimization horizon (2 ms) while respecting all constraints.

Its suboptimality in 1) the assumption of a polynomial output, 2) the restrictive constraint parameterization (of 2%), and 3) the use of LP instead of QP is analyzed.

A cascaded control structure is chosen [19]. The speed loop is controlled via a PI controller, and the current (and torque) loop is the predictive controller. The electrical subsystem of a non-salient PMSM, consisting of the torque-generating and field-generating currents i_q and i_d (peak values), and the machine torque τ_M as output, is given as

$$\frac{\mathrm{d}}{\mathrm{dt}} i_d = -\frac{R}{L} i_d + n_p \omega_M i_q + \frac{1}{L} v_d \tag{27}$$

$$\frac{\mathrm{d}}{\mathrm{dt}} i_q = -\frac{R}{L} i_q - n_p \omega_M i_d - n_p \omega_M K + \frac{1}{L} v_q \tag{28}$$

$$\tau_M = \frac{3}{2} n_p K i_q \tag{29}$$

which is linearized by the assumption that speed is constant over the prediction horizon,

$$\frac{\mathrm{d}}{\mathrm{dt}}\omega_M(t_0) \approx 0 \quad \Rightarrow \omega_M(t) = \omega_M(t_0) \quad \forall t \in [t_0, t_0 + T]. \tag{30}$$

The parameters are taken from a real machine: $R = 0.86$ Ω, $L = 6$ mH, $n_p = 3$, $K = 0.236$ Vs, rated speed 314 rad/s, rated torque 10 Nm. The trajectory optimization shall minimize the control error

$$P_{ctrl}(t) = (\tau_M(t) - \tau_M^*)^2 \tag{31}$$

and at the same time optimize the power efficiency by minimizing the dissipated energy

$$P_{loss}(t) = R\left(i_d^2 + i_q^2\right) + \frac{\omega_M}{R_m}\left((Li_d + K)^2 + i_q^2\right), \tag{32}$$

where $R_m = 1800\Omega$, thus the goal is to minimize $J = \int_0^T (qP_{ctrl}(t) + P_{loss}(t))dt + q\,T\,P_{ctrl}(T)$ with the weight $q = 20$. The flat outputs i_d and i_q are each parameterized with a degree 5 power series. The prediction horizon is $T = 2$ ms and the sampling rate is 10 kHz. The optimization must also maintain the current constraints

$$i_d^2 + i_q^2 \leq I_{max}^2 = 10^2\mathrm{A}^2 \tag{33}$$

to protect the power inverter, and the voltage constraints

$$v_d^2 + v_q^2 \leq V_{max}^2 = 330^2\mathrm{V}^2 \tag{34}$$

to obtain a feasible trajectory. These constraints are linearized as shown in Fig. 3, similar as in [20]. The argumentation is that only $i_d \leq 0$ makes sense (field-weakening, see (32)), and that the current and voltage range in q-axis is more important for PMSMs with isotropic rotors.

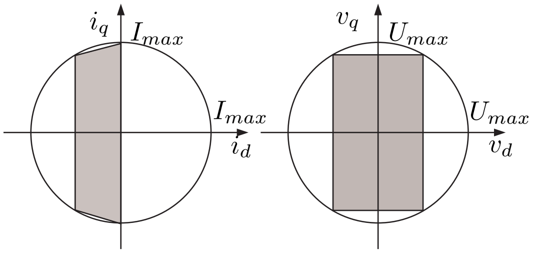

Fig. 3. Linearized current and voltage constraints. Circle: feasible set of current and voltage vectors, grey: feasible set after linearization of the constraints.

Numerical simulation results are shown in Fig. 4. The results of the QP solver are shown in blue, and the results of the LP solver in red. Experimental results are shown in [21]. At $t = 0.01$ s, a speed setpoint change from $\omega_M = 0$ to 420 rad/s is commanded. Then, at $t = 0.07$ s, a load torque step of 8 Nm is applied. From the computational results, QP takes 17 iterations in the worst case, wereas LP takes 24 iterations. This increased number is logical, as there are 20 parameters instead of 10. As the LP algorithm is much simpler, this still represents a considerable computational saving. With an optimized C implementation, the worst case computation time of the LP was 20μs on a PC (1.4 GHz clock). Runtime is further discussed in [21].

From a qualitative standpoint, the resulting behavior is identical for both solvers. The system is always operating at its performance limit. There is a (feasible) voltage peak on v_q at $t = 0.01$ s to rapidly establish a torque current i_q, and after this, the induced voltage increases proportional to speed on v_d and v_q. The iron losses are reduced by imposing $i_d < 0$. Then, at high speed, the load step is quickly compensated via the cascaded speed controller, which commands a torque increase. To do this, the trajectory optimization generates a negative peak on i_d (green arrow) which, according to the model, reduces the induced voltage on i_q, thereby $\frac{\mathrm{d}}{\mathrm{dt}}i_q$ is higher as when keeping i_d small. This is the advantageous behavior of predictive control; in contrast to feedback controllers with saturation, the coupling between the states is exploited to bypass the constraints in an optimal fashion. This result is only visible as a high optimization horizon and the constraints are included in the trajectory generation.

As the constraint handling of LP is suboptimal, there is a quantitative difference, especially in dynamical operation. The peaks are generally somewhat smaller, and at $t = 0.07$, the negative i_d-peak is of shorter duration and thus less effective.

V. Conclusion

A trajectory optimization scheme minimizing a quadratic cost functional to generate continuous trajectories for a linear control system with linear constraints was presented. The scheme recombines several methods in order to obtain a computationally efficient solution. These are the use of differential flatness and of a parameterization using a polynomial basis. A result on polynomial nonnegativity is used as a suitable way to parameterize the constraints. Further developments are done to formulate a linear programming problem.

Numerical simulations of a predictive controller for a permanent magnet synchronous machine show that the suboptimality of the method is acceptable. The optimization problem is sufficiently simplified and solvable in real-time, in the presented application, 2 ms prediction are implemented at a sampling rate of 10 kHz.

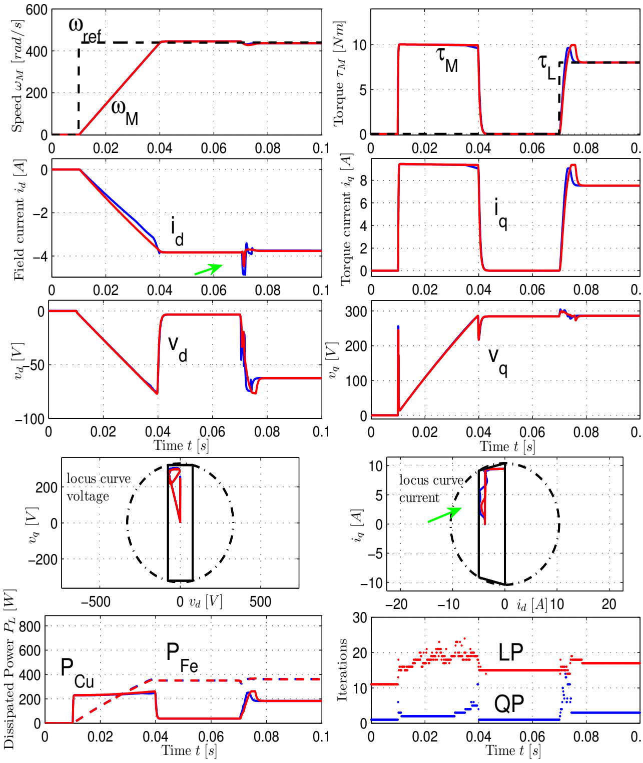

Fig. 4. Results of predictive control applying the presented trajectory optimization. Red: results with LP solver, Blue: results with QP solver.

REFERENCES

[1] M. Fliess and R. Marquez, Continuous-time linear predictive control and flatness: A module-theoretic setting with examples, *Int. Journal of Control*, Vol. 73, No. 7, pp. 606–623, 2000.

[2] D.A. Pierre, *Optimization theory with applications*. New York: Wiley, 1969.

[3] M. Guay and J.F. Forbes, Real-time dynamic optimization of controllable linear systems, *Journal of Guidance, Control and Dynamics*, Vol. 29, No. 4, pp. 929–935, 2006.

[4] N. Petit, M.B. Milam and R.M. Murray, Inversion based constrained trajectory optimization, *Proc. 5th IFAC Symp. on nonlinear control systems design*, 2001.

[5] M.J. van Nieuwstadt and R.M. Murray, Real-time trajectory generation for differentially flat systems, *Int. J. of Robust and Nonlinear Control*, Vol. 8, No. 11, pp. 995–1020, 1998.

[6] K. Graichen and N. Petit, Incorporating a class of constraints into the dynamics of optimal control problems, *Optim. Control Appl. Methods*, Vol. 30, No. 6, pp. 537–561, 2009.

[7] C.V. Rao and J.B. Rawlings, Linear programming and model predictive control, *J. of Process Control*, vol. 10, pp. 283–289, 2000.

[8] M. Fliess, J. Lévine, P. Martin and P. Rouchon, Flatness and defect of nonlinear systems: Introductory theory and examples, *Int. J. of Control*, Vol. 61, No. 6, pp. 1327–1361, 1995.

[9] H. Sira-Ramírez and S.K. Agrawal, *Differentially flat systems*. New York: Marcel Dekker, 2004.

[10] L.V. Kantorovich and V.I. Krylov, *Approximate methods of higher analysis*. New York: Interscience, 1964.

[11] H.R. Sirisena, Computation of optimal controls using a piecewise polynomial parameterization, *IEEE Trans. Automat. Control*, Vol. 18, No. 4, 1973.

[12] H.R. Sirisena and F.S. Chou, Convergence of the control parameterization Ritz method for nonlinear optimal control problems, *J. of Optim. Theory and Appl.*, Vol. 29, No. 3, pp. 369–382, 1979.

[13] W.E. Bosarge, O.G. Johnson, R.S. McKnight and W.P. Timlake, The Ritz-Galerkin procedure for nonlinear control problems, *SIAM Journal on Numerical Analysis*, Vol. 10, No. 1, pp. 94–111, 1973.

[14] R.E. Brown and M.A. Stone, On the use of polynomial series with the Rayleigh-Ritz method, *Compos. Struct.*, Vol. 39, No. 3-4, pp. 191–196, 1997.

[15] J-F. Stumper and R. Kennel, Inversion of linear and nonlinear observable systems with series-defined output trajectories, *Proc. of the IEEE Int. Symp. on Computer-Aided Control System Design*, pp. 1993–1998, 2010.

[16] D. Henrion and J.-B. Lasserre, LMIs for constrained polynomial interpolation with application in trajectory planning, *Systems & Control Letters*, Vol. 55, No. 6, pp. 473–477, 2003.

[17] R. Fletcher, *Practical methods of optimization*. New York: Wiley, 1987.

[18] P. Cortes, M. Kazmierkowski, R. Kennel, D. Quevedo and J. Rodriguez, Predictive control in power electronics and drives, *IEEE Trans. Ind. Electron.*, Vol. 55, No. 2, pp. 4312–4324, 2008.

[19] E. Delaleau and A.M. Stanković, Flatness-based hierarchical control of the PM synchronous motor, *Proc. of the American Control Conf.*, pp. 65–70, 2004.

[20] S. Bolognani, S. Bolognani, L. Peretti and M. Zigliotto, Design and implementation of model predictive control for electrical motor drives, *IEEE Trans. Ind. Electron.*, Vol. 56, No. 6, pp. 1925–1936, 2009.

[21] J-F. Stumper, A. Dötlinger, J. Jung and R. Kennel, Predictive control of a permanent magnet synchronous machine based on real-time dynamic optimization, *European Conf. on Power Electronics and Appl.*, 2011.

APPENDIX I
PROOF OF PROPOSITION **III.2**

Convexity of the conditioned cost function.
Assume $\mathbf{Q}$ is positive definite. Then,

$$\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) > 0, \ \ \forall \mathbf{x}(t) \neq 0 \tag{35}$$

and

$$J = \int_0^T \mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t)dt > 0, \ \ \mathbf{x}(t) \neq 0 \forall t \in [0, T]. \tag{36}$$

The inverse model replaces $\mathbf{x}(t)$ by polynomials $\mathbf{p}(t)$ with linear coefficients in $\boldsymbol{\alpha}$

$$J = \int_0^T \mathbf{p}^T(t)\mathbf{Q}\mathbf{p}(t)dt > 0, \ \ \mathbf{x}(t) = \mathbf{p}(t) \neq 0 \forall t \in [0, T]. \tag{37}$$

Define the primitives $\mathbf{P}(t) = \int \mathbf{p}(t)dt$ thus $\mathbf{P}(T) = \int_0^T \mathbf{p}(t)dt$

$$J = \mathbf{P}^T(T)\mathbf{Q}\mathbf{P}(T) > 0, \ \ \forall \mathbf{P}(T) \neq 0. \tag{38}$$

The expression $\mathbf{P}(t)$ is the primitive of a polynomial, thus a polynomial, with linear coefficients in $\boldsymbol{\alpha}$, and $\mathbf{P}(T)$ is rewritten as

$$\mathbf{P}(T) = \mathbf{A}\boldsymbol{\alpha} \tag{39}$$

and we assume $\operatorname{rank}(\mathbf{A}) = n$ with $n = \dim(\mathbf{x})$. It follows

$$J = \boldsymbol{\alpha}^T\mathbf{A}^T\mathbf{Q}\mathbf{A}\boldsymbol{\alpha} > 0, \ \ \forall \mathbf{P}(T) \neq 0. \tag{40}$$

The matrix of the parameterized cost function is $\mathbf{K} = \mathbf{A}^T\mathbf{Q}\mathbf{A}$ and as we assumed $\mathbf{Q}$ is positive definite we know $\mathbf{Q} = \mathbf{B}^T\mathbf{B}$ (Cholesky decomposition). The weight matrix is then

$$\mathbf{K} = \mathbf{A}^T\mathbf{B}^T\mathbf{B}\mathbf{A} = (\mathbf{B}\mathbf{A})^T(\mathbf{B}\mathbf{A}) \tag{41}$$

which is positive definite as any matrix $\mathbf{K} = \mathbf{C}^T\mathbf{C}$ for some $\mathbf{C}$ with $\operatorname{rank}(\mathbf{C}) = n$ is positive semidefinite [C.D. Meyer, Matrix analysis and applied linear algebra, SIAM books, 2000, pp. 566]. The parameterized cost functional J is thus a convex function of the parameters $\boldsymbol{\alpha}$. ■

APPENDIX II
PROOF OF PROPOSITION **III.3**

Sufficient affine conditions for the constraints.
The polynomial

$$P(s) = \sum_{i=0}^{N} c_i s^i \leq 0, \tag{42}$$

with $c_i \in \mathbb{R}$, is analyzed on non-positivity over a segment $s \in [0, 1]$. A first necessary and sufficient condition is

$$P(0) = c_0 \leq 0 \tag{43}$$

which in the following is assumed satisfied. Furthermore, the N conditions

$$P\left(\frac{k}{N}\right) \leq 0, \ \ k = 1..N \tag{44}$$

are also assumed to hold for all c_i.

These conditions can be rewritten in matrix notation

$$\mathbf{c}_0 + \mathbf{Q}\mathbf{c} \leq 0 \tag{45}$$

with $\mathbf{c}_0 = (c_0, .., c_0)^T \in \mathbb{R}^N$, $\mathbf{c} = (c_1, .., c_N)^T \in \mathbb{R}^N$ and $\mathbf{Q} \in \mathbb{R}^{N \times N}$ such that

$$\mathbf{Q} = (q_{ij}) = \left(\frac{\partial}{\partial c_j} P(i/N)\right) = \left(\left(\frac{i}{N}\right)^j\right), \\ i = 1..N, j = 1..N. \tag{46}$$

It can be shown that $\det(\mathbf{Q}) \neq 0$ for $N > 0$, and that $\mathbf{Q}$ is positive definite. It follows

$$\mathbf{c} \leq -\mathbf{Q}^{-1}\mathbf{c}_0 \tag{47}$$

which can be placed into the polynomial equation

$$P(s) = c_0 + \mathbf{s}^T\mathbf{c} \leq c_0 - \mathbf{s}^T\mathbf{Q}^{-1}\mathbf{c}_0 = (-c_0)\epsilon \tag{48}$$

with $\mathbf{s} = (s, .., s^N)^T \in \mathbb{R}^N$ and $\epsilon = -1 + \mathbf{s}^T\mathbf{Q}^{-1}(1, .., 1)^T$. As we assumed $c_0 \leq 0$, the upper bound of $P(s)$ under the mentioned conditions is at when ϵ is at its maximum. It can be shown that the upper bound of ϵ, $\Delta = \sup\{\epsilon\}$ $\forall s \in [0, 1]$, is positive and only dependent on N, as $\mathbf{Q}$ is known. Some values, which were computed numerically, are shown in the table below.

N	2	3	4	10	20
$\Delta = \sup\{\epsilon\}$	0.125	0.064	0.041	0.012	0.005

Therefore if the conditions (44) hold, we have

$$P(s) \leq -\Delta P(0). \tag{49}$$

Shifting the conditions by the constant (and negative) factor $\Delta P(0)$, the sufficient conditions for non-positivity of the polynomial $P(s)$ are found:

$$P(0) \leq 0, \tag{50}$$

$$P(\frac{k}{N}) - \Delta \cdot P(0) \leq 0, \ \ k = 1..N. \tag{51}$$

■