# Time-Optimal Nonlinear Model Predictive Control with Minimal Control Interventions

Christoph Rösmann, Artemi Makarow, Frank Hoffmann and Torsten Bertram

IEEE Conference on Control Technology and Applications (CCTA), Kohala Coast, Hawai'i, 2017

# Time-Optimal Nonlinear Model Predictive Control with Minimal Control Interventions

Christoph Rösmann, Artemi Makarow, Frank Hoffmann and Torsten Bertram

***Abstract*— This paper presents a novel approach for time-optimal model predictive control. In contrast to a global uniform time scaling, the underlying optimal control problem rests upon a dynamic, local temporal discretization of the shooting grid. The approach seeks for a grid partition with minimum overall transition time. Furthermore, a multi-stage optimization iteratively adapts the number of grid points during runtime to achieve a minimum number of control interventions. A comparative analysis with previous approaches for three nonlinear control problems demonstrates the superiority of the proposed scheme. The feasibility is experimentally demonstrated for position control of a servo drive operated at 200 Hz.**

## I. INTRODUCTION

Model predictive controllers explicitly consider constraints on control and state variables as they repeatedly solve a finite horizon optimal control problem during runtime [1]. Within the past decade, the interest in efficient numerical realizations of (nonlinear) model predictive control (MPC) has grown considerably to expand their application to the control of nonlinear systems with fast dynamics. The majority of recent approaches focus on direct methods that solve the underlying optimal control problem by means of a finite parameter nonlinear program. In contrast, classical indirect optimal control methods rely on the calculus of variations. Direct methods are divided into two common strategies: sequential (single-shooting) and simultaneous (multiple-shooting, collocation). Sequential methods determine the optimal control by solving the nonlinear program w.r.t. only discretized control inputs. They require the repeated simulation of the future state evolution at each solver step. On the other hand, simultaneous methods discretize both states and controls for optimization. The resulting nonlinear program is of higher dimension but due to its sparse structure exhibits better convergence properties. In particular, multiple-shooting partitions the time horizon into multiple discrete intervals, for which isolated initial value problems are solved simultaneously [2]. Diehl et al. present the real-time iteration scheme which applies multiple-shooting to subsequently refine previous solution at runtime [3]. The underlying nonlinear program is solved with a single sequential-quadratic-programming (SQP) step at each sampling interval. For quadratic programming, the sparse structure of the nonlinear program is exploited by (block) condensing techniques, e.g. in [4]. Other methods apply interior-point methods that exploit the sparse structure [5], [6], [7]. Graichen et al. present an efficient real-time capable MPC method based on projected gradients in [8]. A two-stage transformation technique with interior penalties is applied to nonlinear MPC in [9] to solve an unconstrained auxiliary MPC problem using an efficient gradient method.

In the context of time-optimal MPC, explicit cost terms for minimization of the transition time are considered in [10] for a quasi time-optimal control of a spherical robot. The approach rests upon an indirect solution of the optimal control problem. Lam et al. present an approach for following a reference path in minimal time and achieve time-optimality for sufficiently large time horizons [11]. In applications such as race car automatic control, MPC methods minimize the lap time [12], [13]. Verschueren et al. compute time-optimal motions along a Cartesian path for robotic manipulators [14]. Time-transformation is applied to the underlying time-optimal control problem to specify a fixed integration grid. Van den Broeck et al. propose a method called TOMPC for minimizing the transition time of a point-to-point transition [15]. In an outer optimization layer, the horizon length ($\propto$ transition time) of the control sequence is reduced sequentially until the inner standard quadratic form optimal control problem becomes infeasible. The initial horizon length largely determines the required number of iterations in the outer loop time horizon reduction.

Our previous work [16] introduces an approach to time-optimal nonlinear MPC for point-to-point transitions based on the concept of *Timed-Elastic-Bands* (TEB) [17]. The TEB explicitly incorporates the global temporal resolution as a parameter of optimization. Hereby, the optimal control problem is solved with collocation and full discretization. In contrast to previous approaches, time-scaling becomes obsolete. Nevertheless, the numerical properties are similar to time-scaling with a uniform grid. The approach further maintains the number of discretized states to adhere to the desired discretization width.

The time-optimal controls of many systems are either bang-bang or belong to a small finite set of alternative piecewise constant controls. For these types of problems, the number of effective control interventions at which $u$ changes is significantly smaller than the temporal discretization of the MPC. Therefore the optimization step of the previous problem formulation [16] is overburdened by an excess of redundant control parameters. The same observation applies to time-transformation methods in time-optimal control due to their fixed discretization grids. This paper reformulates the time-optimal point-to-point MPC in a manner that explicitly

This work is funded by the German Research Foundation (DFG, BE 1569/13-1). The authors are with the Institute of Control Theory and Systems Engineering, Technische Universität Dortmund, D-44227, Germany, christoph.roesmann@tu-dortmund.de

achieves a minimal intervention control. For this purpose a dynamic shooting grid represents the time-optimal sequence with significantly fewer effective control parameters compared to previous approaches. The contribution is two-fold: First, the time-optimal control sequence is determined with a dynamic grid and second, the grid size is adapted to seek for the optimal partition with the minimum number of interventions.

The next section presents the time-optimal control approach with a dynamic shooting grid. Section III evaluates and analyzes the approach on two nonlinear benchmark problems and reports experimental results of closed-loop position control of a servo drive. Finally, section IV summarizes the results and provides an outlook on further work.

## II. Time-Optimal Model Predictive Control

### A. Formulation of the Optimal Control Problem

A nonlinear, time-invariant dynamical system with time-dependent state vector $\mathbf{x}(t) \in \mathbb{R}^p$ and control input $\mathbf{u}(t) \in \mathbb{R}^q$ is defined by:

$$\dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big), \quad \mathbf{x}(t=0) = \mathbf{x}_s. \tag{1}$$

The optimization task involves the planning of the control trajectory $\mathbf{u}(t)$ for (1) to transit from the initial state $\mathbf{x}_s$ to the final target state $\mathbf{x}_f$ in minimum time $T$. Consequently, the optimization requires the ongoing solution of a boundary value problem on the time interval $[0, T]$. According to the multiple-shooting technique [2] the overall interval is partitioned into $n$ subintervals:

$$\begin{aligned} 0 = t_0 \leq t_0 + \Delta T_0 \quad &= t_1, \\ t_1 \leq t_1 + \Delta T_1 \quad &= t_2, \\ \vdots \quad & \\ t_{n-1} \leq t_{n-1} + \Delta T_{n-1} \quad &= t_n = T. \end{aligned} \tag{2}$$

In contrast to conventional approaches, individual time intervals $\Delta T_k \in \mathbb{R}_0^+$ for $k = 0, 1, \ldots, n-1$ are retained as explicit parameters that form the sampling intervals of the dynamic shooting grid. The control input trajectory $\mathbf{u}(t)$ is constant during each time interval $\Delta T_k$:

$$\mathbf{u}(t) := \mathbf{u}_k \quad \text{for} \quad t \in [t_k, t_k + \Delta T_k]. \tag{3}$$

Multiple-shooting relies on the solution of isolated initial value problems for each time interval $\Delta T_k$ which later on are included in the optimal control problem as a system of algebraic equations respectively constraints. Thus, so-called shooting nodes $\mathbf{s}_k := \mathbf{x}(t_k)$ are introduced for states on the grid $t_k$. The initial value problem on interval $[t_k, t_k + \Delta T_k]$ with system (1), control $\mathbf{u}_k$ and initial state $\mathbf{x}(t_k) = \mathbf{s}_k$ becomes:

$$\mathbf{x}(t_k + \Delta T_k; \mathbf{s}_k, \mathbf{u}_k) = \int_{t_k}^{t_k + \Delta T_k} \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}_k\big) \mathrm{d}t. \tag{4}$$

For the sake of readability, notation $\mathbf{x}(t; \mathbf{s}_k, \mathbf{u}_k)$ emphasizes the associated shooting node $\mathbf{s}_k$ and control input $\mathbf{u}_k$. Conditions for the existence of a unique solution of (4) are addressed in Carathéodory's existence theorem. We assume that $\mathbf{f}(\cdot)$ is continuous and Lipschitz in $\mathbf{x}(t)$. To account for a feasible transition between $\mathbf{s}_0$ and $\mathbf{s}_n$ the final state at each interval has to coincide with the subsequent shooting node, in particular $s_{k+1} = \mathbf{x}(t_k + \Delta T_k; \mathbf{s}_k, \mathbf{u}_k)$ holds for all $k$.

The time-optimal control problem for the point-to-point transition from $\mathbf{x}_s$ to $\mathbf{x}_f$ is defined as follows:

$$\tilde{V}_n^* = \min_{\mathbf{s}_k, \mathbf{u}_k, T} \int_{t=0}^{T} 1 \, \mathrm{d}t = \min_{\mathbf{s}_k, \mathbf{u}_k, T} T \tag{5}$$

subject to

$$\begin{aligned} &\mathbf{s}_0 = \mathbf{x}_s, \quad \mathbf{s}_n = \mathbf{x}_f, \\ &\mathbf{s}_{k+1} = \mathbf{x}(t_{k+1}; \mathbf{s}_k, \mathbf{u}_k), \\ &\mathbf{g}(\mathbf{s}_k, \mathbf{u}_k) \geq \mathbf{0} \qquad\qquad (k = 0, 1, \ldots, n-1). \end{aligned}$$

$\tilde{V}_n^*$ denotes the minimum objective function value and coincides with the optimal transition time $T^*$. Subscript $n$ indicates the number of intervals respectively controls $\mathbf{u}_k$. Initial $\mathbf{s}_0$ and final shooting node $\mathbf{s}_n$ are constrained by $\mathbf{x}_s$ and $\mathbf{x}_f$. Inequality constraints $\mathbf{g} : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}^r$ are imposed on control input $\mathbf{u}_k$ and shooting node $\mathbf{s}_k$. Constraints often refer to saturation limits on states and controls. Usually, for solving optimal control problems such as (5), the systems dynamic equations are scaled by $T$: $\dot{\mathbf{x}}(t) = T\mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big)$. Consequently, interval $[0, T]$ is mapped to $[0, 1]$, thus making the underlying grid fixed and independent of $T$ (e.g. see [14]).

The dynamic grids (2) dependency on $\Delta T_k$ is incorporated into the optimal control problem by means of the modified nonlinear program:

$$V_n^* = \min_{\mathbf{s}_k, \mathbf{u}_k, \Delta T_k} \sum_{k=0}^{n-1} \Delta T_k \tag{6}$$

subject to

$$\begin{aligned} &\mathbf{s}_0 = \mathbf{x}_s, \quad \mathbf{s}_n = \mathbf{x}_f, \quad 0 \leq \Delta T_k \leq \Delta T_{\max}, \\ &\mathbf{s}_{k+1} = \mathbf{x}(t_k + \Delta T_k; \mathbf{s}_k, \mathbf{u}_k), \\ &\mathbf{g}\big(\mathbf{s}_k(\Delta T_k), \mathbf{u}_k\big) \geq \mathbf{0} \qquad\qquad (k = 0, 1, \ldots, n-1). \end{aligned}$$

Parameter $\Delta T_{\max}$ defines an upper bound on the $k$-th interval length.

*Assumption 1:* Inequality constraints $\mathbf{g}\big(\mathbf{s}_k(\Delta T_k), \mathbf{u}_k\big)$ are satisfied for all intermediate states and controls on the interval $\Delta T_k$ and hence $\mathbf{s}_k$ can be substituted by any $\mathbf{x}(t)$ from $t \in [t_k, t_{k+1}]$ without violating the corresponding inequality constraints in (5).

The assumption ensures that inequality constraints are independent of varying interval lengths $\Delta T_k$. Remark 1 provides a further discussion.

*Assumption 2:* There exists a finite $n > 0$ for which optimal control problem (5) is feasible, and its solution constitutes a unique minimizer such that necessary and sufficient conditions hold.

This assumption is fundamental for every direct optimal control formulation. The reader is referred to [19], [20] for further details.

*Theorem 1:* Let $\mathbf{x}^*(t)$ and $\mathbf{u}^*(t)$ denote the feasible and optimal state and control trajectories solving problem (5) for $t \in [0, T^*]$ according to Assumption 2 and let $\tilde{V}_n^*$ denote the corresponding global minimum value of the objective function. Hereby, system (1) is fully discretized with $n$ piecewise constant control inputs to ensure sufficient degrees of freedom. Moreover, let inequality constraints satisfy Assumption 1 for all $k$. Then, the minimizer of optimal control problem (6) coincides with the minimizer of (5) for $\lambda = 0$, $\Delta T_{\max} \to \infty$ and same $n$ such that $V_n^* = \tilde{V}_n^*$.

*Proof:* Both (5) and (6) share the same $n$ (full discretization). Each grid partition $[t_k, t_k + \Delta T_k]$ in (6) can be represented by an auxiliary time-optimal control problem (5) with a single control input (n=1). Bellman's principle of optimality (Bellman's equation) postulates that the minimum of the concatenation (sum) of all individual time intervals is identical to (6). In case the partitioning is not unique (e.g. two consecutive controls are identical), Assumption 1 ensures that the inequality constraints are independent of the grid discretization. □

*Remark 1:* If constraints depend only on control inputs (e.g. in the case of control bounds) Assumption 1 is fulfilled since $\mathbf{u}(t)$ is constant for each $\Delta T_k$ (see (3)). For constraints involving the state evolution, this property depends on the system equations (1). However, the assumption might be satisfied by oversampling $\Delta T_k$ sufficiently and by including the resulting intermediate solutions of the initial value problems (4) as additional constraints. From a practical and implementational point of view, each $\Delta T_k$ is divided into $m$ uniform time intervals of length $\Delta t_k = \Delta T_k / m$. For intermediate states $\mathbf{x}(t_k + l\Delta t_k; \mathbf{s}_k, \mathbf{u}_k)$ for $l = 1, 2, \ldots, m-1$ further inequality constraints are added. Note, cases $l = 0$ and $l = m$ are already included in (6). To comply with Shannon's sampling theorem the sampling times are bounded by a finite $\Delta T_{\max}$.

*Remark 2:* The reformulation of the nonlinear program (6) retains the local structure of the optimal control problem, since $\Delta T_k$ only affects the two consecutive shooting nodes $\mathbf{s}_k$ and $\mathbf{s}_{k+1}$ explicitly. On the other hand, the optimization benefits substantially from the dynamic grid structure. The transition between two consecutive controls $\mathbf{u}_k$ and $\mathbf{u}_{k+1}$ is denoted as an *intervention* in the control sequence if $\mathbf{u}_k \neq \mathbf{u}_{k+1}$. If the number of time intervals $n$ in (6) (respectively the degrees of freedom in the control sequence) is larger than the minimum number of actually required interventions $n^*$, $n - n^*$ time interval parameters become redundant due to Bellman's principle of optimality and might be reduced to achieve shorter computation times. Furthermore, solvers are required to handle underconstrained problems, e.g. in terms of regularization.

Note, in the previous work [16] a uniform grid is utilized regarding a single temporal parameter $\Delta T$ subject to optimization.

### B. Minimum Control Sequence Representation

Time-optimal control usually operates the plant at either the state or controls saturation limits. For many common control systems and tasks, the optimal trajectory consists of a finite concatenation of a few constant controls $\mathbf{u}_k$ that include controls $\mathbf{u}_{\min}$, $\mathbf{u}_{\max}$ (bang arcs) or controls from the interior (singular arc). This *bang-bang property* is generally proven for single-input nonlinear control-affine systems in the plane with bounds only on controls [18]. Many subsequent publications in the literature exploit this property for the synthesis of control tasks for different systems.

In the following, problem (6) with its dynamic shooting grid is solved in a multi-stage optimization to seek for a minimum control representation, ideally but not necessarily composed of bang and singular arcs. First, the influence of the grid resolution respectively the length of the control sequence $n$ is investigated.

Let $n^*$ denote the length related to the minimum number of required control interventions for a system (1) to transit from $\mathbf{x}_s$ to $\mathbf{x}_f$ in minimum time $T = T^* = V_{n^*}^*$. Moreover, the length to provide a (sub-)optimal but still feasible trajectory is denoted by $n_{\text{crit}} \leq n^*$ and leads to $V_{n_{\text{crit}}}^* \geq V_{n^*}^*$. Consequently, for any $n < n_{\text{crit}}$ problem (6) is infeasible due to the lack of sufficient degrees of freedom and the corresponding objective function is defined as $V_n^* := \infty$. For $n > n^*$ the problem is over-determined. This observation suggests adjusting the length in an iterative manner by comparing the objective functions $V_{n-1}^*$, $V_n^*$ and $V_{n+1}^*$ for a given $n$. The following procedure performs this comparison:

1: **procedure** MULTISTAGESTEP($\mathbf{b}_n$)
2: $\{V_n^*, \mathbf{b}_n^*\} \leftarrow$ SOLVENLP($\mathbf{b}_n$) ▷ solve (6)
3: **if** $V_n^*$ is feasible **then**
4: $\mathbf{b}_{n-1} \leftarrow$ RESIZE($\mathbf{b}_n, n-1$)
5: $\{V_{n-1}^*, \mathbf{b}_{n-1}^*\} \leftarrow$ SOLVENLP($\mathbf{b}_{n-1}$)
6: **if** $V_{n-1}^* \leq V_n^*$ **then**
7: **return** $\mathbf{b}_{n-1}^*$
8: $\mathbf{b}_{n+1} \leftarrow$ RESIZE($\mathbf{b}_n, n+1$)
9: $\{V_{n+1}^*, \mathbf{b}_{n+1}^*\} \leftarrow$ SOLVENLP($\mathbf{b}_{n+1}$)
10: **if** $V_{n+1}^* < V_n^*$ **then**
11: **return** $\mathbf{b}_{n+1}^*$
12: **if** $V_n^*$ is non-feasible **then**
13: Repeat procedure with significantly larger $n$
14: **return** $\mathbf{b}_n^*$

Hereby, $\mathbf{b}_n$ denotes the current parameter vector subject to optimization with $n$ intervals: $\mathbf{b}_n = [\mathbf{s}_0^\intercal, \mathbf{u}_0^\intercal, \Delta T_0, \mathbf{s}_1^\intercal, \mathbf{u}_1^\intercal, \Delta T_1, \ldots, \mathbf{s}_{n-1}^\intercal, \mathbf{u}_{n-1}^\intercal, \Delta T_{n-1}, \mathbf{s}_n^\intercal]^\intercal$. Function SOLVENLP($\mathbf{b}_n$) solves (6) with $\mathbf{b}_n$ as initial solution and RESIZE($\mathbf{b}_n$,$n_{\text{new}}$) adjusts the length of the parameter vector $\mathbf{b}_n$ by insertion or removal of parameters. A new tuple of time interval and control is inserted in between interval $\max\{\Delta T_k | \forall k\}$. Shooting node and time interval are interpolated with first-order hold and controls with zero-order hold. In order to identify the minimum number of interventions $n^*$ the procedure is invoked repeatedly up to convergence of $n$. Note, the non-feasibility check in line 12 does not trigger in case the initial length $n$ along with the initial solution $\mathbf{b}_n$ are already feasible, in particular, if $n \geq n_{\text{crit}} - 1$. Otherwise a linear ($n = n + 1$)

or more advanced search technique is invoked that increases $n$ adequately.

Note, if $n$ does not reflect full discretization of the system, Theorem 1 does not hold anymore such that the minimal cost of (5) and (6) are not necessarily identical. Simulations and experiments in Section III indicate that the dynamic grid is superior compared to a uniform grid w.r.t. optimality and grid resolution.

### C. Closed-Loop Control

This section describes the integration of the previously defined open-loop optimization with state feedback in order to regulate system (1) to the final target state $\mathbf{x}_f$. The resulting control scheme mimics a shrinking horizon problem since the target state is included from the very beginning. At each sampling interval the algorithm operates according to:

1: **procedure** FEEDBACKCONTROLSTEP($\mathbf{b}_n, \mathbf{x}_s, \mathbf{x}_f$)
2: Initialize or update trajectory
3: $\mathbf{b}^*_{\tilde{n}} \leftarrow$ MULTISTAGESTEP($\mathbf{b}_n$)
4: **return** $\{\mathbf{u}^*_0, \mathbf{b}_{\tilde{n}}\}$ ▷ $\mathbf{u}^*_0$ is the first control in $\mathbf{b}^*_{\tilde{n}}$

The current plant state $\mathbf{x}_s$ for optimization is either directly measurable or estimated by a state observer. $\mathbf{b}_n$ denotes the parameter vector fed back from the previous sampling interval. The start solution is given by a linear interpolation $\mathbf{x}_k = \mathbf{x}_s + kn^{-1}(\mathbf{x}_f - \mathbf{x}_s)$ between the start and final state with zero controls $\mathbf{u}_k = \mathbf{0}$.

In line 2 the initial or previous parameter vector is updated by replacing $\mathbf{s}_0$ and $\mathbf{s}_n$ by the most recent states $\mathbf{x}_s$ and $\mathbf{x}_f$ respectively. Furthermore, if $\Delta T_0$ from the initialization already expired, it is erased together with its related shooting node and control in order to seed the optimizer with a more appropriate warm-start. Afterward procedure MULTISTAGESTEP($\mathbf{b}_n$) returns the optimized trajectories with length $\tilde{n}$. From that the imminent control action $\mathbf{u}^*_0$ is applied as input to the plant.

*Remark 3:* Limiting the degrees of freedom in the control sequence is a common technique to reduce the computational burden (*Move-Blocking MPC*). At the same time, it leads to a reduced feasible set for the solution and usually suboptimal regulations. By utilizing the minimum control representation, our approach preserves optimality and feasibility. Hereby, the advantages emerge particularly for control tasks with bang bang characteristics since only a few control interventions are required. Apparently, MULTISTAGESTEP($\cdot$) usually demands up to three solutions of (6) and it is indeed questionable whether this computational overhead is preferred over a single solution with a fine uniform grid. However, in a multi-threaded system, all stages consisting of resizing and solving each are invoked independently and in parallel. Alternatively, $n^*$ might be determined prior to control. Ongoing research addresses the direct adjustment of the control input sequence respectively time intervals based on the identification and elimination of redundant control inputs at once. In this case solving multiple NLPs in each step becomes obsolete.

Stability properties in the context of MPC are summarized in [20]. This approach focuses on point-to-point transitions towards a fixed final state. In practice, time-optimal controllers are not favored for stabilizing control, such that in the vicinity of the target state the MPC (softly) switches to a conventional quadratic cost function. However, convergence to the final state must be preserved. Open-loop solutions with final state constraints ($\mathbf{s}_n = \mathbf{x}_f$) as in (6) are shown to be stable in absence of disturbances and model mismatch if the very first solution is already feasible [20]. In that case, the above algorithm always maintains an already feasible solution. Note, in case of disturbances new samples might be inserted to retain feasibility.

## III. Evaluation and Experiments

This section analyzes the time-optimal control approach on two simulated nonlinear control problems. The feasibility is validated in experiment for position control of an industrial servo drive.

The underlying optimal control problem (6) is solved with IPOPT [21]. IPOPT implements the interior point method for sparse nonlinear programs, supports warm starts and is written in C++. Alternatively, a (sparse) active-set solver might be utilized to exploit the bang-bang property for warm starting active constraints. For IPOPT, the sparse linear solver HSL-MA57 [22] is targeted for small and medium-sized problems. The solver depends on Jacobian matrices for the objective function and constraints as well as the Hessian of the Lagrangian. For the evaluation, IPOPT is fed with numerically calculated Jacobian and Hessian matrices to avoid convergence effects that tend to emerge in iterative BFGS methods [19]. To take advantage of the sparsity structure for calculating the Jacobians and Hessian, the complete nonlinear program is represented as a (hyper-)graph which vertices denote optimization parameters in $\mathbf{b}_n$ and which edges denote the equality and inequality constraints as well as the summands of the objective functions. Dense block Jacobian and Hessian submatrices are computed edge-wise, and finally all submatrices are combined into the overall sparse Jacobian resp. Hessian matrices. Computations are performed in C++ (PC: 3,4 GHz Intel i7 CPU, Ubuntu).

### A. Free-Space Rocket System

This section investigates time-optimal control of the free-space rocket system that constitutes a common benchmark in the MPC literature. Rocket motion and mass are given by:

$$\begin{aligned} \dot{s}(t) &= v(t), \\ \dot{v}(t) &= (u(t) - 0.02v(t)^2)/m(t), \\ \dot{m}(t) &= -0.01u(t)^2 \end{aligned} \tag{7}$$

in which the state $s(t)$ denotes the distance traveled. The rocket's velocity $v(t)$ is bounded to $-0.5 \leq v(t) \leq 1.7$. The mass $m(t)$ is bounded from below $m(t) \geq 0$. Variable $u(t)$ denotes the control input and is limited to $-1 \leq u(t) \leq 1$. The state space model $\mathbf{f}(\cdot)$ refers to the state vector $\mathbf{x}(t) = [s, v, m]^\intercal$. The inequality constraints are $\mathbf{g}(\mathbf{s}_k, u_k) = [u + 1, -u + 1, v + 0.5, -v + 1.7, m]^\intercal \geq \mathbf{0}$. Note, Assumption (1) is satisfied. Whenever the velocity bound is active between two consecutive shooting nodes the constrained $\dot{v} = 0$ is

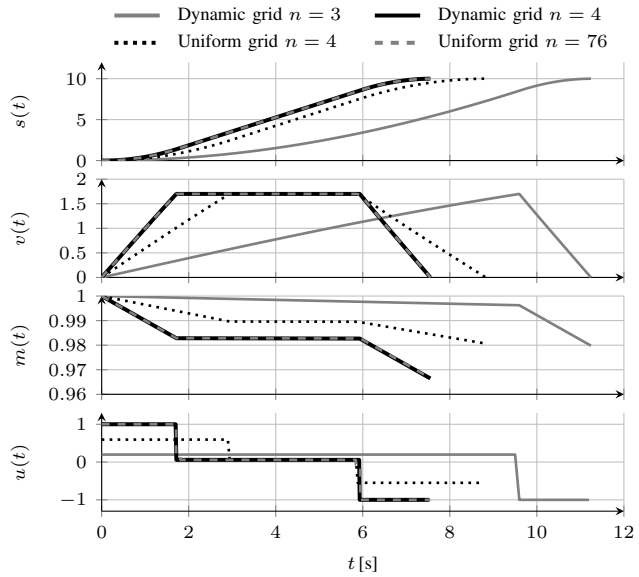


Fig. 1. Open-loop control of the free-space rocket system

imposed which is accomplished by $u_k = 0.02 \cdot 1.7^2 = 0.0578$ in case of the upper bound. This particular $u_k$ is feasible which applies to the other state bounds as well. The control task is to transit from the initial state $\mathbf{x}_s = [0, 0, 1]^\intercal$ to the target state $\mathbf{x}_f = [10, 0, \cdot]^\intercal$ in minimum time. As the final mass $m_f$ is a priori unknown the final state $m$ is subject to optimization. System (7) is integrated with forward Euler and a step width of 0.1s. Figure 1 depicts the resulting state and control input trajectories for different controllers. In this case, the analytical time-optimal control sequence consists of bang and singular arcs. As a reference, the approach presented in [16] with a uniform grid and a single temporal variable subject to optimization are taken into account. To accomplish a comparable resolution, $n = 76$ states are required to maintain a step width of $\Delta T = 0.1$s (full discretization). A uniform grid with fewer states sacrifices upon global optimality (refer to the uniform grid with $n = 4$). In contrast, the dynamic grid approach perfectly coincides with the time-optimal reference control sequence with $n^* = 4$ states and hence three control interventions only. The solution with $n_{\text{crit}} = 3$ is also presented and constitutes a feasible albeit suboptimal solution. Table I reports the corresponding computations times.

TABLE I
COMPUTATION TIME OF ROCKET SYSTEM OPEN-LOOP CONTROL

| Dynamic grid $n = 4$ | Uniform grid $n = 4$ | Uniform grid $n = 76$ |
|---|---|---|
| $(17.2 \pm 0.3)$ ms | $(17.7 \pm 0.5)$ ms | $(66.4 \pm 1.7)$ ms |

### B. Van-der-Pol Oscillator

The Van-der-Pol oscillator constitutes an oscillatory dynamic system with nonlinear damping. The system dynamics equation $\ddot{x}(t) + (x^2(t) - 1)\dot{x}(t) + x(t) = u(t)$ is transformed into a state-space model with the state vector $\mathbf{x} = [x(t), \dot{x}(t)]^\intercal$:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, u) = [\dot{x}, \quad -(x^2 - 1)\dot{x} - x + u]^\intercal. \tag{8}$$

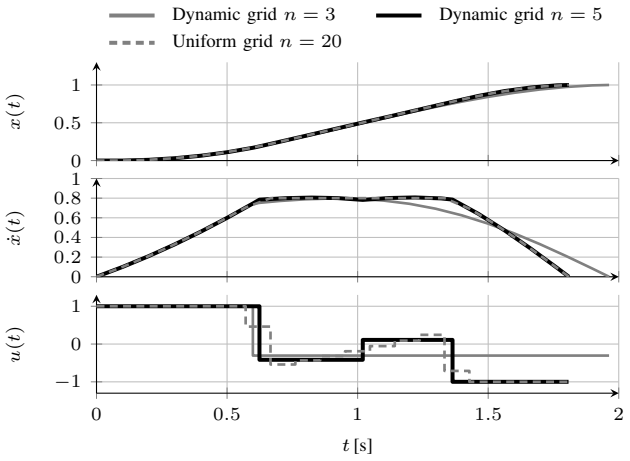


Fig. 2. Open-loop control of the Van-der-Pol Oscillator

The control input $u(t)$ is bounded by $-1 \leq u(t) \leq 1$ and the second state is restricted to $-0.8 \leq \dot{x}(t) \leq 0.8$. Inequality $\mathbf{g}(\cdot)$ is obtained according to the procedure in Section III-A.

The control task demands the transition from an initial state $\mathbf{x}_s = [0, 0]^\intercal$ to the target state $\mathbf{x}_f = [1, 0]^\intercal$ in minimum time. Numerical integration is applied as in section III-A. By adding the state constraint, the second state equation indicates that $-x(t) + u(t) = 0$ must hold for an active bound with $\dot{x}(t) = 0$. Hence, the corresponding control input does not resemble a constant bang or singular arc but rather a linear function dependent of $x(t)$. Furthermore, Assumption 1 might become invalid. In the following, for each interval $\Delta T_k$ 10 intermediate state constraints for $\mathbf{g}(\cdot)$ are added according to Remark 1. Figure 2 shows the resulting state and control trajectories. The reference approach (uniform grid) with $n = 20$ samples exhibits the linear arc in the control sequence for the active state bound. For the dynamic grid, procedure MULTISTAGESTEP is invoked repeatedly with initially $n = 20$ states and eventually converges to $n = 5$. The corresponding input state trajectory merely approximates the linear reference segment and the corresponding state trajectory $\dot{x}(t)$ is slightly curved. However, the transition time for the dynamic grid solution is 1.8093 s whereas the uniform grid achieves 1.8087 s. The small discrepancy is likely attributed to numerical inaccuracies. The figure also illustrates the suboptimal solution for a dynamic grid with $n = 3$.

### C. Closed-Loop Control Experiment

This section investigates closed-loop position control of an *ECP Industrial Plant Emulator Model 220* (see Figure 3). The system consists of two load plates actuated by servo drives which motion is coupled by transmission belts. Drive angles are measured with encoders and the angular velocity is estimated from encoder signals with a DSP.

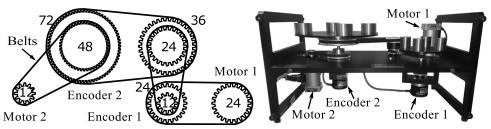


Fig. 3. ECP Industrial Plant Emulator Model 220

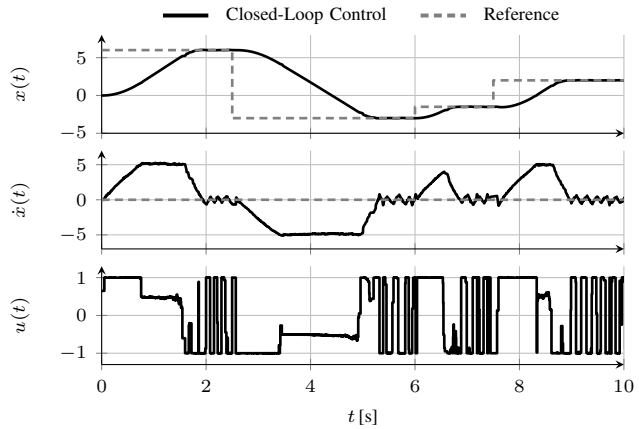

Fig. 4. Closed-loop control of the ECP220 system

In the experimental setup, one plate is the actuator drive which provides a torque via the elastic transmission belt to the second plate which position and velocity are supposed to be regulated by the time optimal controller. The system equation with nonlinear damping is given by:

$$\ddot{x}(t) = 4ku(t)/J - \big(F_c \tanh(\alpha\dot{x}(t) + d\dot{x}(t)\big)/J \quad (9)$$

with $k = 9.5 \cdot 10^{-2}$, $J = 3.39 \cdot 10^{-2}$, $F_c = 9.39 \cdot 10^{-2}$, $\alpha = 5.37$ and $d = 1.93 \cdot 10^{-2}$. The state space model is derived similar to section III-B with state vector $\mathbf{x}(t) = [x(t), \dot{x}(t)]^\intercal$. Furthermore, the control input denotes the current of the actuator drive and is limited to $|u(t)| \leq 1$ while the velocity is bounded to $|\dot{x}(t)| \leq 5$. Note, that the simplified dynamic equations exhibit a non-trivial model mismatch w.r.t. to the true coupled drive dynamics. For the closed-loop control experiment, the reference signal commands four target positions at standstill as shown in Figure 4. The controller is only aware of the current target position (marked with dashed lines as reference) and does not look-ahead beyond the single step reference. Procedure MULTISTAGESTEP converges to $n = 4$ states and hence three control interventions. To accomplish a sampling rate of 200 Hz closed-loop control is performed with the $n = 4$ solution (according to Remark 3). Numerical integration is performed with 5th-order Runge-Kutta with a step width of $0.1$ s. The resulting state and control trajectories are depicted in Figure 4.

## IV. Conclusions and Future Work

The time-optimal optimal control formulation with a dynamic grid achieves time-optimality with minimum burden on the computational resources for two nonlinear systems. In comparison to previous approaches with a uniform grid and a finer discretization, the proposed approach is able to determine the time-optimal control sequence with only a small subset of the parameters. This reduction in problem dimension is especially beneficial for time-optimal control with bang and singular arcs. A multi-stage optimization automatically determines the minimum degrees of freedom either at runtime or a-priori to closed-loop control for slower single-threaded implementations. A closed-loop control example is presented for an experimental test bed that demonstrates the ability to transit the servo drive to different target positions at a sampling rate of 200 Hz.

Future work addresses the development of single-stage optimization algorithms for dynamic-grid based time-optimal MPC as well as its application to hybrid objective functions.

## References


[1] M. Morari and J. H. Lee, "Model predictive control: past, present and future," *Computers & Chemical Engineering*, vol. 23, no. 4-5, pp. 667–682, 1999.

[2] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer, "Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations," *Journal of Process Control*, vol. 12, no. 4, pp. 577–585, 2002.

[3] M. Diehl, H. G. Bock, and J. P. Schlöder, "A real-time iteration scheme for nonlinear optimization in optimal feedback control," *SIAM Journal on Control and Optimization*, vol. 43, no. 5, pp. 1714–1736, 2005.

[4] G. Frison, D. Kouzoupis, J. B. Jørgensen, and M. Diehl, "An efficient implementation of partial condensing for nonlinear model predictive control," in *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 2016.

[5] Y. Wang and S. P. Boyd, "Fast model predictive control using online optimization," in *IFAC World Congress*, vol. 17, 2008, pp. 6974–6979.

[6] A. Richards, "Fast model predictive control with soft constraints," in *European Control Conference*, 2013, pp. 1–6.

[7] K. S. Pakazad, H. Ohlsson, and L. Ljung, "Sparse control using sum-of-norms regularized model predictive control," in *IEEE Conference on Decision and Control*, 2013.

[8] K. Graichen and B. Käpernick, "A real-time gradient method for nonlinear model predictive control," in *Frontiers of Model Predictive Control*, 2012.

[9] B. Käpernick and K. Graichen, "Nonlinear model predictive control based on constraint transformation," *Optimal Control Applications and Methods*, vol. 37, no. 4, pp. 807–828, 2016.

[10] J. Zhao, M. Diehl, R. Longman, H. G. Bock, and J. P. Schlöder, "Nonlinear model predictive control of robots using real-time optimization," in *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 2004.

[11] D. Lam, "A model predictive approach to optimal path-following and contouring control," PhD Thesis, The University of Melbourne, 2012.

[12] D. P. Kelly and R. S. Sharp, "Time-optimal control of the race car: a numerical method to emulate the ideal driver," *Vehicle System Dynamics*, vol. 48, no. 12, pp. 1461–1474, 2010.

[13] J. P. Timings and D. J. Cole, "Minimum manoeuvre time of a nonlinear vehicle at constant forward speed using convex optimisation," in *International Symposium on Advanced Vehicle Control*, 2010.

[14] R. Verschueren, N. van Duijkeren, J. Swevers, and M. Diehl, "Time-optimal motion planning for n-dof robot manipulators using a path-parametric system reformulation," in *Proceedings of the American Control Conference (ACC)*, 2016.

[15] L. Van 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.

[16] C. Rösmann, F. Hoffmann, and T. Bertram, "Timed-elastic-bands for time-optimal point-to-point nonlinear model predictive control," in *European Control Conference (ECC)*, 2015.

[17] C. Rösmann, W. Feiten, T. Wösch, F. Hoffmann, and T. Bertram, "Efficient trajectory optimization using a sparse model," in *European Conference on Mobile Robots*, 2013, pp. 138–143.

[18] H. J. Sussmann, "The structure of time-optimal trajectories for single-input systems in the plane: The general real analytic case," *SIAM Journal on Control and Optimization*, vol. 25, no. 4, pp. 868–904, 1987.

[19] J. Nocedal and S. J. Wright, *Numerical optimization*, ser. Springer series in operations research. New York: Springer, 1999.

[20] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[21] A. Wächter and L. T. Biegler, "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[22] "HSL. A collection of Fortran codes for large scale scientific computation." [Online]. Available: http://www.hsl.rl.ac.uk/