Receding horizon time-optimal control for a class of differentially flat systems

D. Lam\textsuperscript{a}, C. Manzie\textsuperscript{a,}\textsuperscript{*}, M.C. Good\textsuperscript{a}, R.R. Bitmead\textsuperscript{b}

\textsuperscript{a} Department of Mechanical Engineering, The University of Melbourne, Victoria 3010, Australia
\textsuperscript{b} Department of Mechanical & Aerospace Engineering, University of California, San Diego, 9500 Gilman Drive, La Jolla, CA 92093-0411, USA

A R T I C L E   I N F O

Article history:
Received 24 February 2014
Received in revised form 23 April 2015
Accepted 1 July 2015
Available online 18 July 2015

Keywords:
Model predictive control
Time optimal control
Differentially flat systems

A B S T R A C T

A closed-loop, time-optimal path-following control scheme is proposed for a class of constrained differentially flat systems. Within a receding horizon framework, a finite horizon optimisation problem is solved at each sample, using available state feedback and feedforward path information. Irrespective of horizon length, the proposed formulation guarantees exact path-following. Moreover, the requirements under which the proposed algorithm achieves minimum-time path-following are established. Simulations conducted with a rigid X–Y table model confirm the theoretical results.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Many industrial applications require the output of a dynamic system to track a predetermined geometric path. Examples include the control of CNC machines and robotic manipulators. In these applications, it is often desired to traverse the path in minimum time, in order to maximise machine productivity. However, such systems are often subject to actuator constraints which limit their acceleration capability.

A number of approaches for generating time-optimal path-following trajectories for constrained robotic manipulators have been proposed. These include dynamic programming methods\cite{1} and phase plane techniques\cite{2}. In the latter, a numerical search is employed to find the switching points between minimum and maximum path acceleration. Two-pass algorithms based on velocity and acceleration limits were proposed in\cite{3}. In\cite{4}, the minimum-time path-following problem is reformulated as a convex optimisation problem, resulting in efficient calculation of the optimal solution. Differentially flat systems possess a special structure which may be exploited to solve minimum time problems, see for example\cite{5}. In\cite{6}, it is shown that path-following problems involving such systems may be expressed in a form that is useful for analysis. All of these approaches generate open loop solutions which in practice are used as reference trajectories for feedback controllers. As a result, time-optimality is compromised by the inclusion of the lower level feedback controller dynamics.

Several researchers have attempted to address this problem by proposing online trajectory generation schemes. In\cite{7}, the speed of the reference trajectory is adjusted online using trajectory time scaling. The path acceleration and velocity constraints are calculated, and if violation of these constraints is detected, a time scaling factor is applied to slow the reference down. The path governor approach\cite{8} generates a reference trajectory online in a receding horizon fashion. While the path governor attempts to minimise the traversal time, time optimality of the control scheme is not guaranteed. For the online trajectory generation approaches\cite{7,8}, a separate feedback controller is still required to track the reference trajectory.

In contrast, path-following controllers determine the evolution of the path and the plant inputs simultaneously using available feedback. A number of path-following control schemes exist in the literature (see for example\cite{9}), however these approaches do not consider constraints on the system. A constrained model predictive path-following controller is proposed in\cite{10} where a finite horizon optimisation problem is solved in a receding horizon fashion. In the context of contouring control, a similar approach is proposed in\cite{11} where the cost function to be minimised is based on competing objectives of productivity and accuracy. However, conditions under which receding horizon path-following controllers achieve time optimality have yet to be investigated.

\textsuperscript{*} Corresponding author.

E-mail addresses: lamd@unimelb.edu.au (D. Lam), manziec@unimelb.edu.au (C. Manzie), mcgood@unimelb.edu.au (M.C. Good), rbitmead@ucsd.edu (R.R. Bitmead).

http://dx.doi.org/10.1016/j.sysconle.2015.07.001
0167-6911/© 2015 Elsevier B.V. All rights reserved.
In this paper, a closed-loop path-following control strategy is proposed with the intention of generating time-optimal trajectories. Following a receding horizon framework, at each sample an optimisation problem is posed over a finite horizon, whose solution determines both the path evolution and control inputs concurrently. Unlike offline time-optimal control approaches, the proposed receding horizon strategy uses available feedback information at each sample, as well as a preview of the path. Sufficient conditions which guarantee time-optimality of the proposed algorithm are determined; most notably a minimum horizon length is required.

To illustrate the approach, simulations are performed with the algorithm implemented for a two-dimensional robot cutting task. This real-world example conforms to the differential flatness conditions and exhibits the exact achievement of the global time-optimal solution via receding horizon time-optimal control. The non-pathological nature of the control solution is demonstrated along with the graceful approach to the global control. The non-pathological nature of the control solution is flatness conditions and exhibits the exact achievement of the algorithm implemented for a two-dimensional robot cutting task.

2. The minimum-time path-following problem

Consider a continuous time nonlinear system
\[
\dot{\xi}_t = f_t(\xi_t, u_t), \quad q_t = f_q(\xi_t),
\]
where \(\xi_t \in \mathcal{X} \subset \mathbb{R}^n\), \(u_t \in \mathcal{U} \subset \mathbb{R}^m\) and \(q_t \in \mathbb{R}^p\) represent the state, input and output vector respectively, with input and state constraint sets \(\mathcal{U}\) and \(\mathcal{X}\) which contain the respective origin.

The control task is to steer the system output along a regular curve, parametrised by a scalar path parameter \(\theta\) from \(\theta^0\) to \(\theta^f\):
\[
q^f(\theta) : \mathbb{R} \rightarrow \mathbb{R}^p, \quad \theta \in [\theta^0, \theta^f].
\]
(2)
The path function \(q^f(\theta)\) is assumed to be twice differentiable with bounded second derivative with respect to \(\theta\). Given an initial time \(t\), path parameter \(\theta_t\) and state \(\xi_t\), the aim is to traverse forwards along the path to \(\theta^f\) in minimum time, subject to the constraints of the system \((1)\). Define the following dynamics for \(\theta\)
\[
\dot{\theta}_t = v_t, \quad v_t \geq 0,
\]
where \(v_t\) is a virtual input which determines the path evolution. The constraint \(v_t \geq 0\) enforces forward motion along the path. The minimum-time path-following problem can be posed as:
\[
\mathcal{P}_\infty(t, \theta_t, \xi_t) : \text{Minimise } \int_t^{\theta_f} 1 \, d\tau,
\]
Subject to \(\dot{\theta}_t = v_t, \quad v_t \geq 0, \quad \theta_t \in [\theta^0, \theta^f], \quad \xi_t = f_t(\xi_t, u_t), \quad u_t \in \mathcal{U}, \quad \xi_t \in \mathcal{X}, \quad u_t = q_t^f(\theta_t), \quad \forall \tau \in [t, \infty),
\]
where \([u_t, v_t]\) are the control inputs and \(t_f\) is the time taken to complete the path. Note that since the final two constraints in \((4)\) apply for all \(\tau \in [t, \infty)\), the system is forced to steady state at the end of the path.

3. Receding horizon formulation

A closed-loop receding horizon approach is proposed with the intention of achieving minimum time control. The idea is to advance along the path as far as possible within a finite horizon of length \(N\). Consider the following finite horizon optimisation problem:
\[
\mathcal{P}_N(t, \theta_t, \xi_t) : \text{Minimise } - \int_t^{t+N} \theta_t \, d\tau,
\]
Subject to \(\dot{\theta}_t = v_t, \quad v_t \geq 0, \quad \theta_t \leq \theta^f, \quad \xi_t = f_t(\xi_t, u_t), \quad u_t \in \mathcal{U}, \quad \xi_t \in \mathcal{X}, \quad f_q(\xi) = q^f(\theta_t), \quad \forall \tau \in [t, \infty),
\]
where \([u_{\tau}, v_{\tau}]\) are the control inputs and \(t_f\) is the time taken to complete the path. Note that since the final two constraints in \((4)\) apply for all \(\tau \in [t, \infty)\), the system is forced to steady state at the end of the path.

4. Properties of receding horizon time-optimal control

Since the optimisation \(\mathcal{P}_N\) includes output and state constraints, it is important to ensure that recursive feasibility is maintained throughout the trajectory. Define the set of all feasible input trajectories for \(\mathcal{P}_\infty(t, \theta_t, \xi_t)\) and \(\mathcal{P}_N(t, \theta_t, \xi_t)\) as \(\mathcal{X}_\infty(\theta_t, \xi_t)\) and \(\mathcal{X}_N(\theta_t, \xi_t)\) respectively. It is assumed throughout that at \(t = 0\) there exists at least one feasible solution for \((4)\) and \((5)\).

Assumption 1. \(\mathcal{X}_\infty(\theta_0, \xi_0)\) and \(\mathcal{X}_N(\theta_0, \xi_0)\) are non-empty.

The terminal constraint \(f_q(\xi_{t+N}, u_{t+N}) = 0\) ensures that in each optimisation, the state at end of the horizon lies in a control invariant set. Hence, provided that \((5)\) is feasible at initialisation, recursive feasibility of RH-TOC is guaranteed \([12]\).
The finite horizon optimisation problem \( \mathcal{P}_N \) occurs over only a portion of the path. As a result, the question arises of whether or not the RH-TOC Algorithm produces time-optimal trajectories. In this section, properties of differentially flat systems are used to show that for a sufficiently long horizon, \( (u^{001}, v^{001}) \) is time-optimal.

It is assumed that starting from rest at any point on the path, it is possible to complete the path in finite time. This is stated formally in Assumption 2.

**Assumption 2.** For all \( \theta_0 \in [\theta^s, \theta^f] \), if \( q_0 = \dot{q}^d(\theta_0) \) and \( \tau_0(\xi_0, u_0) = 0 \), then \( \mathcal{X}_\infty(\theta_0, \xi_0) \) is non-empty.

Differential flatness is a useful property when considering path-following problems. In particular, time optimality of RH-TOC will be analysed for second order differentially flat systems, as described in the following definition.

**Definition 1.** The system (1) is second order differentially flat if:

1. \( \xi \) can be expressed as a function of \( q \) and \( \dot{q} \),
   \[ \xi = \Phi(q, \dot{q}), \]
2. \( u \) can be expressed as a function of \( q, \dot{q} \) and \( \ddot{q} \),
   \[ u = \Psi(q, \dot{q}, \ddot{q}), \]
3. the elements of \( q \) are differentially independent; they do not satisfy any differential equation.

The analysis for the remainder of this paper is restricted to systems which are second order differentially flat.

**Assumption 3.** The system (1) is second order differentially flat.

**Remark 1.** Rigid fully actuated robotic manipulators with invertible forward kinematics, such as those considered in \([1, 2]\), satisfy Assumption 3. However, second order differentially flat systems need not be mechanical. For example, the Van der Vusse chemical reactor presented in \([6]\) also satisfies Assumption 3.

It is shown in \([6]\) that if Assumption 3 holds, the minimum-time path-following problem \( \mathcal{P}_\infty \) is equivalent to

\[
\mathcal{P}_\infty(t, \theta_1, v_1) : \begin{aligned}
\text{Minimise} & \quad \int_{t}^{t_f} 1 \, dt, \\
\text{Subject to} & \quad \dot{t} = v, \quad v \geq 0, \\
& \quad \theta = \theta, \quad v \geq 0, \\
& \quad u = \tilde{\Psi}(\tilde{\theta}, \dot{\tilde{\theta}}) \in \mathcal{X}, \\
& \quad \tau(\theta, \dot{\theta}) \in \mathcal{X}, \\
& \quad \xi(\tau(\theta, \dot{\theta})) \in \mathcal{X},
\end{aligned}
\]

where

\[
\tilde{\Psi}(\tilde{\theta}, \dot{\tilde{\theta}}) = \Phi(q^d(\theta), \dot{q}^d(\theta), \ddot{q}^d(\theta, \dot{\theta})), \\
\dot{\Phi}(\theta, \dot{\theta}) = \Phi(q^d(\theta), \dot{q}^d(\theta, \dot{\theta})).
\]

In a similar manner, the finite-horizon problem \( \mathcal{P}_N \) can be equivalently expressed as

\[
\mathcal{P}_N(t, \theta_1, v_1) : \begin{aligned}
\text{Minimise} & \quad -\int_{t}^{t+N} \theta \, dt, \\
\text{Subject to} & \quad \dot{\theta} = v, \quad v \geq 0, \\
& \quad \theta = \theta, \quad v = 0, \\
& \quad u = \tilde{\Psi}(\tilde{\theta}, \dot{\tilde{\theta}}) \in \mathcal{X}, \\
& \quad \xi(\tau(\theta, \dot{\theta})) \in \mathcal{X},
\end{aligned}
\]

The optimisation problems \( \mathcal{P}_\infty \) and \( \mathcal{P}_N \) are completely described by two states, \( \theta \) and \( v = \dot{\theta} \), and one input \( \theta \). As a result, the solutions to (8) and (10) can be analysed in the \( \theta-v \) phase plane. This approach was used in \([2, 14]\) to construct open loop solutions to the minimum-time problem \([4]\). The same approach is employed to investigate time optimality of RH-TOC. The following definition will be useful for the analysis.

**Definition 2.** Given a trajectory \( [\theta_1, v_1] \), define \( \tau(\theta) \) and \( v(\theta) \) as follows:

\[
\tau(\theta) = \min\{\tau : \theta = \theta_1\}, \\
v(\theta) = v(\theta_1).
\]

Denote the set of all feasible trajectories for \( \mathcal{P}_\infty(t, \theta_1, v_1) \) and \( \mathcal{P}_N(t, \theta_1, v_1) \) as \( \mathcal{X}_\infty(\theta_1, v_1) \) and \( \mathcal{X}_N(\theta_1, v_1) \) respectively. The following two lemmas will be used later to prove time optimality of RH-TOC.

**Lemma 1.** Let Assumption 3 hold and for any \( (t, \theta_1, \xi_1) \in \mathcal{X}_\infty(\theta_1, v_1) \) non-empty. Then the optimal trajectory for \( \mathcal{P}_\infty(t, \theta_1, \xi_1) \) follows from a similar argument \([3]\). The uniqueness of \( [\theta^*_1, v^*_1] \) follows from a similar argument \([3]\).
Denote optimal input trajectories for (15) as \([u^0_\tau(t), v^0_\tau(t)]\) and the optimal \(t_\tau\) as \(t^0_\tau(t)\), which is the minimum time required to come to rest along the path from initial state \([\theta^*_\tau, \xi^*_\tau]\).

**Condition 1.** The horizon length \(N\) satisfies

\[
N - \Delta t \geq \max_t t^0_\tau(t).
\]

(16)

As shown in the following, **Condition 1** ensures that, starting from an appropriate initial condition, there exists a feasible trajectory for the finite horizon problem \(\mathcal{P}_N\) that coincides with the minimum-time trajectory over an initial interval of length \(\Delta t\).

**Lemma 3.** Let **Condition 1** hold. Then for any \(t\) there exists \([\tilde{u}_\tau, \tilde{v}_\tau] \in \mathcal{X}_N(\theta^*_\tau, \xi^*_\tau)\) such that \(\tilde{v}_\tau = v^*_\tau\) and \(\tilde{u}_\tau = u^*_\tau\) for all \(t \in [t, t + \Delta t]\).

**Proof.** Construct \([\tilde{u}_\tau, \tilde{v}_\tau] = \{u^*_\tau, v^*_\tau\}\) as follows:

\[
[\tilde{u}_\tau, \tilde{v}_\tau] = \begin{cases}
(u^*_\tau, v^*_\tau), & \tau \in [t, t + \Delta t], \\
(v^0_\tau(t + \Delta t), u^0_\tau(t + \Delta t)), & \tau \in (t + \Delta t, t_\tau], \\
(u^0_\tau(t + \Delta t), v^0_\tau(t + \Delta t)), & \tau \in (t_\tau, t + N],
\end{cases}
\]

where \(t_\tau = t + \Delta t + t^0_\tau(t + \Delta t)\). Note that since (16) holds, \(t^0_\tau(t + \Delta t) \leq N - \Delta t\). By the constraints of (15), \(f_\tau(\xi_{t+N}, u_{t+N}) = 0\). Therefore, \([\tilde{u}_\tau, \tilde{v}_\tau] \in \mathcal{X}_N(\tilde{\theta}_\tau, \tilde{\xi}_\tau)\).

The following lemma states that the initial segment of length \(\Delta t\) of \([v^{**, \tau}, u^{**, \tau}]\) is equal to the time-optimal trajectory, and will be used to show that under **Condition 1**, \([u^{**}_\tau, v^{**}_\tau]\) is time-optimal.

**Lemma 4.** Let Assumptions 1–3 and **Condition 1** hold. For any \(t\), let \([u^{**, \tau}, u^{*, \tau}]\) be the optimal input trajectory for \(\mathcal{P}_N(t, \theta^*_\tau, \xi^*_\tau)\). Then the following holds:

\[
u^{*, \tau}(\theta) = v^{**, \tau}(\theta^*_\tau), \quad \forall \theta \in [t, t + \Delta t].
\]

(18)

**Proof.** Since (1) is second order differentially flat, (18) holds if for all \(\theta \in [\theta^*_\tau, \theta^{**, \tau}]\), \(v^*(\theta) = v^{**, \tau}(\theta^*_\tau)\). Assume to the contrary that there exists \(\theta \in [\theta^*_\tau, \theta^{**, \tau}]\) such that \(v^*(\theta) \neq v^{**, \tau}(\theta^*_\tau)\).

Relation (19) contradicts optimality of either \([u^*_\tau, v^*_\tau]\) or \([u^{**, \tau}, v^{**, \tau}]\). Therefore, (18) must hold.

**Theorem 1.** Let Assumptions 1–3 and **Condition 1** hold. Then \([u^{**}_\tau, v^{**}_\tau]\) is the time-optimal solution for \(\mathcal{P}_N(0, \theta_0, \xi_0)\).

**Proof.** It follows immediately from **Lemma 4** that the input trajectory produced by the RH-TOC Algorithm is the time-optimal solution for \(\mathcal{P}_N(t, \theta_0, \xi_0)\).

**Theorem 1** implies that if \(N\) satisfies **Condition 1**, RH-TOC is a closed-loop minimum-time path-following control strategy for second order differentially flat systems. Furthermore, the RH-TOC Algorithm is guaranteed to be recursively feasible regardless of horizon length. The RH-TOC Algorithm may also be applied to systems which are not second order differentially flat. However, as the proofs presented in this paper rely on phase plane analysis, time optimality of the control approach for more general systems remains an open question.

**5. Implementation of the receding horizon time optimal control**

In its present guise, RH-TOC is not necessarily robust to plant variation and disturbance presence; time-optimality, differential flatness, and constraint satisfaction each might cause non-robustness. This is a property per se of the control problem being addressed rather than of any solution method. The contribution of this paper has been to demonstrate that full-horizon time-optimality can be achieved in a feedback form using receding-horizon methods and not to advance a notion that the receding-horizon solution avoids the pitfalls of the full-horizon problem in terms of robustness.

Indeed, the robustness of Model Predictive Control to both model error and/or disturbances is largely an open problem, even under the condition that the full state is measured (which obviates the need to consider model complexity mismatch). Although some headway has been made in constraint tightening to accommodate disturbances [15,16]. Performance aspects of MPC also have only recently been addressed [17,18], but not in a robustness context nor, indeed, in a disturbance rejection framework. It must therefore be assumed that robustness of RH-TOC also will be problematic to establish without resort to ad hoc implementation techniques such as the replacement of hard constraints by penalty functions and the relaxation of time-optimality to bring the input values off their constrained limits or, at least, to restrict the disturbance induced switching between limits.

Practical applications of the RH-TOC framework incorporate appropriate modifications to handle disturbances and modelling errors, but sacrifice theoretical time-optimality. For example, put path-following constraint can be replaced with a penalty on path-following accuracy in the cost function, and if the system satisfies Assumption 3, the terminal constraint can be satisfied by enforcing the input constraint \(v_{t+N} = 0\), avoiding potential feasibility issues. In the resulting optimisation problem, the constraint \(f_q(\xi_t) = q^*(\theta_t)\) is removed and the objective altered to minimise

\[
\int_{t}^{t+N} \left\{ -\dot{\theta} + \| f_\tau(\xi_t) - q^*(\theta_t) \|_{Q} \right\} \, dt.
\]

for \(Q \in \mathbb{R}^{n \times p}\) positive definite and typically large. Inevitably, exact time-optimal path-following is not guaranteed when this modification is applied, however the controller may be tuned in order to achieve a desired level of performance, and feasibility issues arising from plant/model mismatch are avoided. This is demonstrated experimentally in [11], where the penalty weighting on path-following accuracy is further tuned in order to sacrifice accuracy for speed.

**6. Simulation results**

To demonstrate the theoretical results derived in Section 4, simulations were conducted where RH-TOC was implemented on a rigid X–Y table model with the following equation of motion:

\[
\begin{bmatrix}
J_x & 0 \\
0 & J_y
\end{bmatrix} \ddot{q} + \begin{bmatrix}
B_x & 0 \\
0 & B_y
\end{bmatrix} \frac{\dot{q}}{P} = \begin{bmatrix}
0 \\
0
\end{bmatrix}.
\]

(20)

For the \(x\) and \(y\) axes respectively, \(q\) is the vector containing the \(x\) and \(y\) axis positions, \(P\) is the lead screw pitch for both axes, \(u_x, u_y\) are the input torques, \(J_x, J_y\) are rotational inertias, and \(B_x, B_y\) are coefficients of viscous friction. The control inputs are subject to the constraints \(|u_x|, |u_y| \leq 1\) N m. It can easily be shown that the model (20) is a second order differentially flat system. The two paths illustrated in Fig. 1 were considered.

The RH-TOC algorithm was implemented by approximating the system (20) with a discrete-time system and then solving the discrete-time equivalent of \(\mathcal{P}_N\). Fig. 2 shows plots of the input torques \(u_x, u_y\) and virtual input \(v\) produced by the receding horizon controller for the elliptical path. The trajectory was confirmed to be
time-optimal using the results of [2]. As expected of a minimum-time trajectory, at least one of \( u_x \) and \( u_y \) is saturated as the path is being traversed.

In order to investigate the effect of horizon length on traversal time, simulations were also conducted with the flower-shaped path with a range of values of \( N \). Fig. 3 shows plots of path speed and joint velocities versus \( \theta \) for four different horizon lengths. As expected, it can be observed that using shorter horizons limits the joint velocities as the system traverses the path. The relationship between traversal time and horizon length is shown in Fig. 4. The results indicate that the trajectories gracefully approach the minimum-time trajectory as \( N \) increases. Beyond \( N = 0.036 \), there is no further improvement in traversal time, indicating that Condition 1 is satisfied. The selection of the tuning parameter \( N \) can be seen as a tradeoff between traversal time and computational complexity for practical implementations of the algorithm.

Fig. 1. Desired path shapes used for simulation.

Fig. 2. Simulation results for elliptical path.

Fig. 3. \( \dot{x}, \dot{y} \) and \( v \) versus \( \theta \) for flower-shaped path with varying horizon lengths.

Fig. 4. Traversal time versus horizon length for flower-shaped path.
7. Conclusions

A receding horizon path-following algorithm has been developed for a class of differentially flat systems based on the idea of advancing as far along the path as possible within a finite horizon. The closed-loop approach makes use of available feedback at each time step, and is shown to be equivalent to a time-optimal solution with full path information provided that the horizon satisfies a minimum length condition. Further work may include investigating time-optimality for a wider class of systems, implementing efficient solution methods for the finite horizon optimisation and improving robustness of the scheme to disturbances and plant/model mismatch.

References