A Real-Time Energy-Optimal Trajectory Generation Method for a Servomotor System

Wang, Y; Zhao, Y.; Bortoff, S.A.; Ueda, K.

Abstract

This paper considers real-time energy-optimal trajectory generation for a servomotor system which performs a single-axis point-to-point positioning task for a fixed time interval. The servomotor system is subject to acceleration and speed constraints. The trajectory generation is formulated as a linear constrained optimal control problem (LCOCP), and the Pontryagin's Maximum Principle is applied to derive necessary optimality conditions. Instead of solving multi-point boundary value problems directly, this paper proposes a novel real-time algorithm based on two realizations: solving the LCOCP is equivalent to determine an optimal time interval of the speed constrained arc and solve a specific acceleration constrained optimal control problem (ACOCP); solving an ACOCP is equivalent to determine optimal switch times of acceleration constrained arcs and solve a specific two-point boundary value problem (TBVP). The proposed algorithm constructs sequences of time intervals, ACOCPs, switch times, and TBVPs, such that all sequences converge to their counterparts of an optimal solution of the LCOCP. Numerical simulation verifies that the proposed algorithm is capable of generating energy-optimal trajectories in real-time. Experiments validate that the use of energy-optimal trajectories as references in a servomotor system does not compromise tracking performance but leads to considerable less energy consumption.

A Real-Time Energy-Optimal Trajectory Generation Method for a Servomotor System

Yebin Wang, *Member, IEEE,* Yiming Zhao, *Member, IEEE,* Scott A. Bortoff, *Senior Member, IEEE,* and Koichiro Ueda

Abstract—This paper considers real-time energy-optimal trajectory generation for a servomotor system which performs a single-axis point-to-point positioning task for a fixed time interval. The servomotor system is subject to acceleration and speed constraints. The trajectory generation is formulated as a linear constrained optimal control problem (LCOCP), and the Pontryagin's Maximum Principle is applied to derive necessary optimality conditions. Instead of solving multi-point boundary value problems directly, this paper proposes a novel real-time algorithm based on two realizations: solving the LCOCP is equivalent to determine an optimal time interval of the speed constrained arc and solve a specific acceleration constrained optimal control problem (ACOCP); solving an ACOCP is equivalent to determine optimal switch times of acceleration constrained arcs and solve a specific two-point boundary value problem (TBVP). The proposed algorithm constructs sequences of time intervals, ACOCPs, switch times, and TBVPs, such that all sequences converge to their counterparts of an optimal solution of the LCOCP. Numerical simulation verifies that the proposed algorithm is capable of generating energy-optimal trajectories in real-time. Experiments validate that the use of energy-optimal trajectories as references in a servomotor system does not compromise tracking performance but leads to considerable less energy consumption.

Index Terms—Servomotor, motion planning, trajectory generation, minimum energy, constrained optimal control, boundary value problem.

I. INTRODUCTION

SERVOMOTOR systems are typically used for motion control and have a wide range of applications including robotic manipulators, electromechanical systems such as milling machines, cranes, lathes. In these applications, a servomotor system performing a point-to-point positioning task prompts several problems: path planning and trajectory generation [1]–[3], tracking control [4], motor control [5]–[7] and steady state optimization [8] etc. This paper focuses on the trajectory generation problem, i.e., the servomotor follows a well-defined path. More specifically, the servomotor system performs point-to-point tasks in a single axis. Solving this problem on real-time is non-trivial due to various constraints and severe restriction on the computation time.

For decades, intensive efforts have been devoted to the trajectory generation problem, and lead to numerous results, for instance, time optimal or approximate time optimal trajectory [3], [9]–[12], trapezoidal speed trajectory [13], energy-optimal trajectory [14], [15], maximum payload trajectory [16]. Time optimal trajectories lead to maximal productivity, but sacrifice energy efficiency. Recent trends on building greener industries and widespread uses of mobile devices have attracted much attention on trajectory generation to balance productivity and energy consumption e.g. [17]–[21]. Work [19] investigated the minimization of motor's resistant losses using optimal control techniques. Neither the friction effect nor the mechanical work is taken into account. Based on an optimal control formulation, work [20], [21] proposed real-time minimum-energy trajectory generation algorithms for mobile robots. Nevertheless, work [19]–[21] did not address speed and acceleration constraints. A three phase trapezoidal speed trajectory has been widely used for energy efficiency, e.g. [13]. This method is capable of dealing with both acceleration and speed constraints, and is appropriate for real-time applications, albeit the resultant trajectory is sub-optimal. More recently, [22] used genetic algorithms to generate trajectories which minimize resistant losses of a motor-toggle servomechanism performing position-to-position tasks. This method is capable of addressing various constraints, but is not intended for real-time applications.

This paper aims to develop real-time energy-optimal trajectory generation algorithms for a servomotor system subject to physical constraints. Main challenges come form the demanding computation time target and physical constraints. The main contributions of this paper are as follows: first, a real-time algorithm is proposed for energy-optimal trajectory generation; second, the convergence of the proposed algorithm is established; finally, experiments are performed to verify the energy-saving advantage of using energy-optimal trajectories in a servomotor system. The proposed algorithm can be readily implemented in motion controllers for real-time energy-optimal trajectory generation. The rest of this paper is organized as follows. Problem formulation is provided in Section II. Section III presents algorithms for real-time computation of energy-optimal trajectories. Numerical results are given in Section IV to validate the proposed algorithm and its computational efficiency. Experimental results in Section V demonstrate the energy-saving benefits of using energy-optimal trajectories in

Y. Wang is with Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA 02139 USA (Phone: +1 617 621-7500, fax: +1 617 621-7550, email: yebinwang@ieee.org).

Y. Zhao was with Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA 02139 USA. He is with Halliburton Energy Service, 3000 N Sam Houston Pkwy E, Houston, TX 77032 USA (email: yiming.zhao@halliburton.com).

S. A. Bortoff is with Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA 02139 USA (email: bortoff@merl.com).

K. Ueda is with the Advanced Technology R&D Center, Mitsubishi Electric Corporation, 8-1-1, Tsukaguchi-honmachi, Amagasaki City, 661–8661, Japan (email: Ueda.Koichiro@da.MitsubishiElectric.co.jp).

a servomotor system. Section VI concludes this paper.

II. PROBLEM STATEMENT

For a servomotor system performing point-to-point positioning tasks in a single axis, its simplified dynamics are given by

$$J\ddot{x}(t) = -d_0 v(t) - c_0 \text{sgn}(v) + K_t u(t),$$

where x is the angular position of the servomotor, $v = \dot{x}$ is the angular speed, u is the control current, J is the lumped inertia of the servomotor and an inertia load, c_0 is the amplitude of the Coulomb friction force, d_0 is the viscous friction coefficient, and K_t is the torque constant. Note that in this paper the coulomb friction force is simply modeled as: $-c_0 \text{sgn}(v)$ for $v \neq 0$, otherwise zero. Also, since the servomotor always rotates in one direction for each specific positioning task, without loss of generality, we consider $v \geq 0$ and have the Coulomb friction force given by $-c_0$. For notational convenience, let $d = d_0/J$, $c = c_0/J$ and $b = K_t/J$. The servomotor system dynamics are rewritten as

$$\begin{aligned} \dot{x}(t) &= v(t), \\ \dot{v}(t) &= -dv(t) - c + bu(t). \end{aligned} \quad (1)$$

The servomotor system is subject to the following speed and acceleration constraints:

$$0 \leq v \leq v_{\max}, \quad (2a)$$

$$A_{\min} \leq -dv - c + bu \leq A_{\max}, \quad (2b)$$

where $v_{\max}$, $A_{\min} < 0$, and $A_{\max} > 0$ are constants. The cost functional, representing the energy consumption of (1), captures the copper loss and mechanical work, i.e.,

$$E = \int_0^{t_f} (Ru^2 + K_t vu)\mathrm{d}t, \quad (3)$$

where R is the motor resistance, and t_f is the final time.

Remark 2.1: Equation (3) is a simplification of [15, Eqn. (3)]. While the servomotor system decelerates, $P(x, u) = Ru^2 + K_t vu$ could be negative, i.e., mechanical work is converted into electricity. Taking (3) as a cost function means that the regenerated electricity during deceleration is recycled.

The energy-optimal trajectory generation problem is formulated as follows.

Problem 1: Given the servomotor system (1), constraints (2), the initial state $(0, 0)$, the final state $(x_f, 0)^T$, and the final time t_f, find (x^*, v^*, u^*) which minimize the cost function (3), i.e.,

$$\begin{aligned} &\min_u & E &= \int_0^{t_f} P(v(t), u(t))\mathrm{d}t \\ &\text{subject to} & &(1) \text{ and } (2), \\ & & &x(0) = v(0) = v(t_f) = 0, \ x(t_f) = x_f. \quad (4) \end{aligned}$$

Without loss of generality, we assume that $x_f > 0$.

Remark 2.2: Problem 1 is a linear constrained optimal control problem (LCOCP), and can be solved using numerical optimization methods [23]. Alternative approaches include applying the Pontryagin's Maximum Principle (PMP) and its variants, e.g. [24], to derive necessary optimality conditions (NOCs), which are further reduced to Multi-point Boundary

Value Problems (MBVPs). Work [15] generalizes the conventional PMP to deal with switching cost functions, and solves the resultant nonlinear constrained optimal control problem by reducing NOCs to MBVPs. Interested readers are referred to [15] and references therein for details. Generic solvers for MBVPs admit large scale nonlinear programming problems (NLPs) thus have difficulty in meeting the computation time target for real-time applications. Another disadvantage of resorting to an NLP formulation is lack of convergence results. Our main goal is to alleviate these limitations and develop reliable as well as computationally efficient algorithms for solving Problem 1 in real-time.

Remark 2.3: For every positioning task, there might exist an optimal t_f^* giving the best energy efficiency. The optimal t_f^* can be obtained by solving Problem 1 with the free final time cost function $E = \int_0^{t_f} P(v(t), u(t))\mathrm{d}t$.

A. Existence, Arcs and Smoothness

The analysis of existence, NOCs, and structures of optimal solutions in [15] is also applicable to Problem 1. One can conclude that Problem 1 has an optimal solution, and apply the PMP and its extensions to derive NOCs. Interested readers are referred to [15] for details. Define a speed constrained arc as a segment of an optimal trajectory, where $v(t) = v_{\max}$, i.e., the speed constraint is active. Similarly, the activation of constraints $\dot{v} \leq A_{\max}$ and $\dot{v} \geq A_{\min}$ corresponds to acceleration and deceleration constrained arcs, respectively. From NOCs of Problem 1, one knows that optimal trajectories of Problem 1 may contain as many as four types of arcs: the acceleration constrained arc, the deceleration constrained arc, the speed constrained arc, and the unconstrained arc.

The connection points between neighboring arcs are called *junction points*. A *corner points* (or simply *corners*) appears when the control is discontinuous. For Problem 1, because the Hamiltonian is strictly convex with respect to the control, the optimal control is continuous along an optimal trajectory [24, Prop. 4.3]. That is: an optimal trajectory does not contain any corner point, or the optimal speed trajectory for Problem 1 is $\mathcal{C}^1$. The continuous differentiability of the optimal speed trajectory is essential to design computationally efficient algorithms for energy-optimal trajectory generation.

III. OPTIMAL SOLUTIONS AND ALGORITHMS

Although the same approach in [15], [25] is followed to solve Problem 1, this paper proposes a real-time algorithm which comprises the following key steps

- define a linear unconstrained optimal control problem (LUOCP), and find the optimal solution of each LUOCP by solving the associated Two-point Boundary Value Problem (TBVP);
- define an acceleration constrained optimal control problem (ACOCP), and construct sequences of switch times and LUOCPs to solve the ACOCP;
- construct sequences of time intervals of speed constrained arcs and ACOCPs to solve Problem 1;
- construct an optimal solution of Problem 1 from optimal solutions of LUOCPs and constrained arcs.
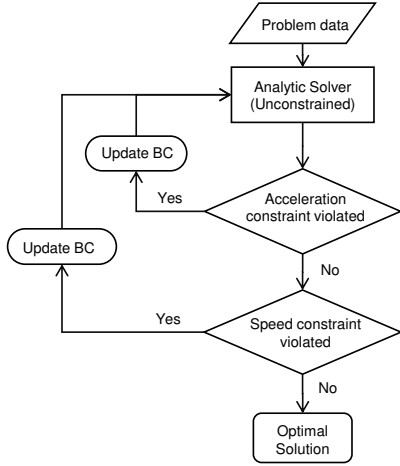
Fig. 1. The flow chart of the proposed algorithm.

Fig. 1 illustrates the flow chart of the proposed algorithm, where BC stands for boundary conditions. The analytical solver block computes the optimal solution of an LUOCP. At each iteration of the inner loop, a new LUOCP is constructed with different BC, which is updated by examining the optimal solution of the current LUOCP against the acceleration constraints (2b). At each iteration of the outer loop, a new ACOCP is constructed with different BC, which is updated by examining the optimal solution of the current ACOCP against the velocity constraints (2a).

A. Solve Linear Unconstrained Optimal Control Problems

This section presents and analyzes the analytical solution of the LUOCP as follows.

Problem 2: Given the servomotor system (1), the initial state $(0,0)$, the final state $(x_f, 0)^T$, and the final time t_f, find (x^*, v^*, u^*) which minimize the cost function (3), i.e.,

$$\min_{u} \quad E = \int_0^{t_f} P(v(t), u(t)) \mathrm{d}t$$
$$\text{subject to} \quad \text{(1) and (4)}.$$

Without loss of generality, the LUOCP takes the same BC as Problem 1. When applying the PMP to solve Problem 2, we first derive NOCs, then construct and solve the TBVP corresponding to the NOCs. Define the Hamiltonian

$$H(x(t), v(t), \lambda_x(t), \lambda_v(t), u(t)) =$$
$$Ru^2(t) + K_t v(t)u(t) + \lambda_x(t)v(t) + \lambda_v(t)(bu(t) - dv(t) - c),$$

where the co-state λ_x and λ_v have dynamics

$$\dot{\lambda}_x(t) = -\frac{\partial H}{\partial x} = 0, \tag{5a}$$

$$\dot{\lambda}_v(t) = -\frac{\partial H}{\partial v} = d\lambda_v(t) - K_t u(t) - \lambda_x(t). \tag{5b}$$

According to (5a), $\lambda_x(t)$ is constant. The optimal control u^* shall satisfy the first-order optimality condition $\partial H/\partial u = 0$, and be taken the following expression

$$u(t) = -\frac{K_t}{2R}v(t) - \frac{b}{2R}\lambda_v(t). \tag{6}$$

The optimal control u is uniquely determined because of the convexity of H with respect to u. The NOCs of Problem 2 consist of (1), (5), (6), and the BC (4). Substituting (6) into (1) and (5) yields a linear TBVP

$$\dot{x}(t) = v(t),$$
$$\dot{v}(t) = -\left(d + \frac{bK_t}{2R}\right)v(t) - \frac{b^2}{2R}\lambda_v(t) - c,$$
$$\dot{\lambda}_v(t) = \frac{K_t^2}{2R}v(t) + \left(d + \frac{bK_t}{2R}\right)\lambda_v(t) - \lambda_x, \tag{7}$$
$$x(0) = 0, \ x(t_f) = x_f, \ v(0) = 0, \ v(t_f) = 0.$$

where λ_x, $\lambda_v(0)$ and $\lambda_v(t_f)$ are unknown. Denote $X = (x, v, \lambda_v)^T$. Differential equations in (7) can be written as

$$\dot{X} = AX + B, \tag{8}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & A_1 & A_2 \\ 0 & A_3 & -A_1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ -c \\ -\lambda_x \end{bmatrix},$$

with $A_1 = -(d + \frac{bK_t}{2R})$, $A_2 = -\frac{b^2}{2R}$, $A_3 = \frac{K_t^2}{2R}$. The analytic solution to (8) is given by

$$X(t) = M(t, t_0)X(t_0) + G(t, t_0)B, \tag{9}$$

where $M(t, t_0) = e^{A(t-t_0)}$ and $G(t, t_0) = \int_{t_0}^t e^{A(t-\tau)}B\mathrm{d}\tau$. Considering (9), the BC takes the following linear expression

$$X(t_f) = M(t_f, 0)X(0) + G(t_f, 0)B. \tag{10}$$

where three unknowns $\lambda_v(0)$, $\lambda_v(t_f)$, and λ_x can be solved. The optimal solution to the TBVP (8) and the optimal control can be determined from (9) and (6), respectively.

The following two propositions characterize important properties of the energy-optimal speed trajectory for Problem 2.

Proposition 3.1: Let $(x, v, \lambda_x, \lambda_v)$ and $(\tilde{x}, \tilde{v}, \tilde{\lambda}_x, \tilde{\lambda}_v)$ be two optimal solutions of Problem 2 with different BC and/or final time t_f. If v and $\tilde{v}$ are not identical and define $z(t) = v(t) - \tilde{v}(t)$, then

1) the equation $z(t) = 0$ has at most two roots;
2) if $\mathcal{S} = \{t|z(t) = 0\}$ is non-empty, equation $\dot{z}(t) = 0$ has at most one root.

Proof: 1). Without loss of generality, let $t_0 = 0$ be a root of $z(t) = 0$, then $v(0) = \tilde{v}(0)$. From (9), $v(t)$ is given by

$$v(t) = M_{2,1}(t)x(0) + M_{2,2}(t)v(0) + M_{2,3}(t)\lambda_v(0)$$
$$- cG_{2,2}(t) - \lambda_x G_{2,3}(t),$$

where $M_{i,j}$ and $G_{i,j}$ are the elements in the ith-row, and j^{th}-column of matrices M and G, respectively. Specifically, $M_{2,1}(t) \equiv 0$, and the above equation can be simplified to

$$v(t) = M_{2,2}(t)v(0) + M_{2,3}(t)\lambda_v(0) - cG_{2,2}(t) - \lambda_x G_{2,3}(t). \tag{11}$$

Similarly, we have

$$\tilde{v}(t) = M_{2,2}(t)\tilde{v}(0) + M_{2,3}(t)\tilde{\lambda}_v(0) - cG_{2,2}(t) - \tilde{\lambda}_x G_{2,3}(t). \tag{12}$$

Subtracting (11) from (12) gives

$$z(t) = M_{2,3}(t)(\tilde{\lambda}_v(0) - \lambda_v(0)) - (\tilde{\lambda}_x - \lambda_x)G_{2,3}(t)$$

$$= M_{2,3}(t)\Delta\lambda_{v_0} - \Delta\lambda_x G_{2,3}(t).$$

Let $\tau \neq 0$ be a non-zero root of $z(t) = 0$, then τ must satisfy

$$M_{2,3}(\tau)\Delta\lambda_{v_0} - \Delta\lambda_x G_{2,3}(\tau) = 0,$$

which can be further simplified according to the analytic expressions of $M(t)$ and $G(t)$:

$$\Delta\lambda_{v_0} = \Delta\lambda_x \left(\frac{2A_3}{A_2 p}\right)\tanh(p\tau/4), \tag{13}$$

where $p = 2\sqrt{A_1^2 + A_2 A_3}$.

If $\tilde{\lambda}_x = \lambda_x$, $\Delta\lambda_x = 0$, then (13) holds only if $\tilde{\lambda}_v(0) = \lambda_v(0)$, which further implies that $\tilde{v}(t) \equiv v(t)$ for any t according to (11) and (12)–a contradiction. Therefore, $\tilde{\lambda}_x \neq \lambda_x$, $\Delta\lambda_x \neq 0$, and (13) can be written as

$$\frac{\Delta\lambda_{v_0}}{\Delta\lambda_x} = \frac{2A_3}{A_2 p}\tanh(p\tau/4), \tag{14}$$

Since the left hand side (LHS) of (14) is a constant, and the right hand side (RHS) is a strictly monotonically increasing function, (14) has at most one solution. Hence, $z(t)$ has at most one root other than $t = 0$. The proof of 1) is completed.

2). By differentiating (11) and (12) with respect to t and subtracting one from the other, we obtain

$$\dot{z}(t) = \Delta\lambda_{v_0} A_2 \cosh(pt/2) + \frac{2\Delta\lambda_x A_2}{p}\sinh(pt/2)$$
$$= \frac{2\Delta\lambda_x A_2}{p}\cosh(pt/2)\left(\frac{p\Delta\lambda_{v_0}}{2\Delta\lambda_x} + \tanh(pt/2)\right).$$

Since $\cosh(pt/2) \geq 1$, and $\tanh(pt/2)$ is strictly monotonic, $\dot{z}(t) = 0$ has at most one solution. 2) is proven. ∎

Proposition 3.2: Let $(x, v, \lambda_x, \lambda_v)$ be the optimal solution of Problem 2. The following facts hold

1) there exist $t_1, t_2 \in \mathbb{R}$ such that $\dot{v}(t_1)\dot{v}(t_2) < 0$;
2) $\dot{v}(t)$ is a strictly monotonic function.

Proof: 1). It is clear that $v(t)$ is continuously differentiable over $[0, t_f]$. To ensure $x_f > 0$, it is necessary to have T_1 such that $v(T_1) > 0$. Considering the fact $v_0 = 0$, there exists $t_1 \in [0, T_1]$ such that $\dot{v}(t_1) > 0$. On the other hand, the facts $v_f = 0$ and $v(T_1) > 0$ implies the existence of $t_2 \in [T_1, t_f]$ such that $\dot{v}(t_2) < 0$. The proof of 1) is complete.

2). From (8), we have

$$\dot{v}(t) = A_1 v(t) + A_2 \lambda_v(t) - c. \tag{15}$$

Eliminating $v(t)$ from the RHS of (15) by considering (9) and the expressions of $M(t, t_0)$ and $G(t, t_0)$, we have

$$\dot{v}(t) = (A_2\lambda_{v_0} - c + A_1 v_0)\cosh(pt/2)$$
$$+ 2(v_0 A_1^2 - cA_1 - A_2\lambda_x + A_2 A_3 v_0)\sinh(pt/2)/p.$$

Let $\tau = pt/2$. Noticing that $\sinh(\tau) = (e^\tau - e^{-\tau})/2$ and $\cosh(\tau) = (e^\tau + e^{-\tau})/2$, there exist constants c_1, $c_2 \in \mathbb{R}$ such that $\dot{v}(t)$ can be rewritten as

$$v'(\tau) = c_1 e^\tau + c_2 e^{-\tau},$$

where v' denotes the derivative of v with respect to τ. Besides, with $\tau_1 = pt_1/2$ and $\tau_2 = pt_2/2$, we have $v'(\tau_1)v'(\tau_2) < 0$. Without loss of generality, assume that $\tau_1 < \tau_2$. Since v'

is proportional to $\dot{v}$, it suffices to prove that v' is a strictly monotone function.

If $v'(\tau_1) > 0$ and $v'(\tau_2) < 0$, then we must have $c_1 c_2 < 0$, otherwise, $v'(\tau)$ is either always positive or always negative, which leads to a contradiction. Now consider the second derivative of v: $v''(\tau) = c_1 e^\tau - c_2 e^{-\tau}$. Note that $v''(\tau)$ is continuous. Because $c_1 c_2 < 0$, we must have either $v''(\tau) > 0$ for any $\tau \in \mathbb{R}$ when $c_1 > 0$, or that $v''(\tau) < 0$ for any $\tau \in \mathbb{R}$ when $c_1 < 0$, which implies that $v'(\tau) > 0$ is either a strictly monotonically increasing function when $c_1 > 0$ or a strictly monotonically decreasing function when $c_1 < 0$. Similar conclusion holds when $v'(\tau_1) < 0$ and $v'(\tau_2) > 0$. Therefore, the proof is complete. ∎

Remark 3.3: Proposition 3.2 means that given an optimal solution of Problem 2 (x^*, v^*, u^*), the largest acceleration and deceleration $\dot{v}^*(t)$ appear at $t = 0$ and $t = t_f$, respectively.

B. Solve Acceleration Constrained Optimal Control Problems

We are ready to solve the following ACOCP.

Problem 3: Given the servomotor system (1) subject to the acceleration constraints (2b), the initial state $(0, 0)$, the final state $(x_f, 0)^T$, and the final time t_f, find (x^*, v^*, u^*) which minimize the cost function (3), i.e.,

$$\min_u \quad E = \int_0^{t_f} P(v(t), u(t))\mathrm{d}t$$
$$\text{subject to} \quad (1), (2b), \text{ and } (4).$$

Remark 3.4: The ACOCP is different from Problem 2 by including acceleration constraints (2b). Work [15, Sec. 4] provides a complete analysis of possible structures of optimal solutions for Problem 1. It is not difficult to conclude that an optimal solution for Problem 3 in general exhibits a three-phase structure: an acceleration constrained arc in the first phase, followed by an unconstrained arc, finally a deceleration constrained arc in the third phase. The three-phase structure is illustrated by Fig. 2, where three phases are defined over $[t_0, t_1], [t_1, t_2], [t_2, t_f]$, respectively. Such a three-phase structure is also inferred from Remark 3.3.

An optimal solution of the ACOCP may not contain the first and/or the third phase, depending on problem data including $A_{\max}$ and $A_{\min}$, model parameters, initial and final states, and the final time t_f. We however focus on the general case and assume that the optimal solution has all three phases. Given t_1 and t_2, the optimal solution in the first and third phases can be determined explicitly as follows

$$v_l(t) = A_{\max}t, \quad t \in [0, t_1], \tag{16a}$$

$$x_l(t) = \frac{1}{2}A_{\max}t^2,, \quad t \in [0, t_1], \tag{16b}$$

$$v_r(t) = A_{\min}(t - t_f), \quad t \in [t_2, t_f], \tag{16c}$$

$$x_r(t) = x_f + \frac{1}{2}A_{\min}(t - t_f)^2, \quad t \in [t_2, t_f]. \tag{16d}$$

Denoting $X_m = (x_m, v_m, u_m)^T$ the optimal solution in the second phase defined over $[t_1, t_2]$, we have

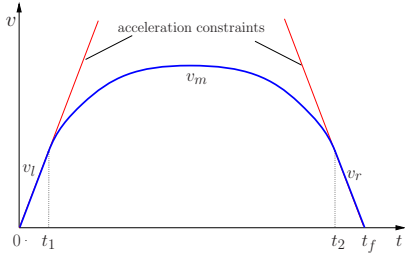$$X_m(t) = M(t, t_1)X_m(t_1) + G(t, t_1)B. \tag{17}$$

Fig. 2. Optimal speed trajectory with active acceleration constraints.

Since no constraint is active in the second phase, the optimal solution $X_m(t)$ over $[t_1, t_2]$ can be obtained by solving Problem 2 with the following BC:

$$
\begin{aligned}
x_m(t_1) &= x_l(t_1), & v_m(t_1) &= v_l(t_1), \\
x_m(t_2) &= x_r(t_2), & v_m(t_2) &= v_r(t_2).
\end{aligned}
\tag{18}
$$

We use the term ATBVP to represent the TBVP defined by (1), (5), (6), and the BC (18).

Remark 3.5: The definition of the ATBVP is contingent on switch times t_1 and t_2. This is because the BC (18) has explicit dependence on t_1 and t_2. As shown in [15], switch times can be determined from switch conditions which are part of NOCs. For Problem 3, the optimal control is continuous, and switch conditions are given by $u(t_1^+) = u(t_1^-)$ and $u(t_2^+) = u(t_2^-)$, or equivalently

$$
\dot{v}_l(t_1^-) = \dot{v}_m(t_1^+), \quad \dot{v}_m(t_2^-) = \dot{v}_r(t_2^+).
\tag{19}
$$

Clearly, the ACOCP is reduced to find t_1 and t_2 satisfying switch conditions (19), and solve the resultant ATBVP.

Remark 3.6: An MBVP formulation for the ACOCP contains a total of thirteen equations–seven algebraic equations (16)-(17), four BCs (18), two switch conditions (19)–and thirteen unknowns $(t_1, t_2, x_m(t_1), x_m(t_2), v_m(t_1), v_m(t_2), \lambda_v(t_1), \lambda_v(t_2), x_l(t_1), v_l(t_1), x_r(t_2), v_r(t_2), \lambda_x)$. Hence, the MBVP is solvable. On the other hand, the MBVP is nonlinear thus difficult to solve efficiently, because (16b) and (16d) are nonlinear, and switch times t_1 and t_2 enter nonlinearly into (17) through $M(t_2 - t_1)$ and $G(t_2 - t_1)$ terms, and are multiplied by other unknowns. We focus on algorithms to solve the MBVP efficiently and reliably.

We propose Algorithm 1 to solve the ACOCP by identifying optimal switch times t_1, t_2. Once the optimal switch times t_1 and t_2 are obtained, the optimal solution of Problem 3 can be computed by combining (16) and the solution of the ATBVP. Algorithm 1 conducts the following steps in an iterative manner

1) given switch times (t_1, t_2), constructs an ATBVP defined by (1), (5), (6), and the BC (18);
2) solves the optimal solution of the ATBVP;
3) updates the switch times (t_1, t_2) according to the optimal solution and switch conditions.

Fig. 3 briefly illustrates the construction of sequences of t_1 and t_2 at the kth iteration of Algorithm 1. More details can be found in Algorithm 1. The convergence of Algorithm 1 is established by the following theorem. The proof of Theorem 3.7 is given in Appendix.

Algorithm 1 Algorithm for Problem 3

1: $e \Leftarrow 1$, $k \Leftarrow 1$, and choose a small tolerance parameter $\varepsilon \ll 1$
2: $t_{a_k} \Leftarrow 0$, $t_{b_k} \Leftarrow t_f$
3: **while** $e > \varepsilon$ **do**
4:     solve the unconstrained problem with BC $v_k(t_{a_k}) = v_l(t_{a_k})$, $x_k(t_{a_k}) = x_l(t_{a_k})$, $v_k(t_{b_k}) = v_r(t_{b_k})$, $x_k(t_{b_k}) = x_r(t_{b_k})$, and denote the solution as $(x_k, v_k, u_k)^T$.
5:     **if** $\dot{v}_k(t_{a_k}) > A_{\max}$ **then**
6:         solve $\dot{v}_k(\tau_l) = A_{\max}$ for τ_l
7:         $t_{a_{k+1}} \Leftarrow \tau_l$
8:     **else**
9:         $t_{a_{k+1}} \Leftarrow t_{a_k}$
10:     **end if**
11:     **if** $\dot{v}_k(t_{b_k}) < A_{\min}$ **then**
12:         solve $\dot{v}_k(\tau_r) = A_{\min}$ for τ_r
13:         $t_{b_{k+1}} \Leftarrow \tau_r$
14:     **else**
15:         $t_{b_{k+1}} \Leftarrow t_{b_k}$
16:     **end if**
17:     $k \Leftarrow k + 1$
18:     $e \Leftarrow |t_{a_k} - t_{a_{k-1}}| + |t_{b_k} - t_{b_{k-1}}|$
19: **end while**
20: $t_1 \Leftarrow t_{a_k}$, $t_2 \Leftarrow t_{b_k}$
21: **return** x^*, v^*, u^*, where

$$
x^*(t) = \begin{cases} x_l(t), & 0 \le t \le t_1, \\ x_k(t), & t_1 < t \le t_2, \\ x_r(t), & t_2 < t \le t_f, \end{cases}
$$

$$
v^*(t) = \begin{cases} v_l(t), & 0 \le t \le t_1, \\ v_k(t), & t_1 < t \le t_2, \\ v_r(t), & t_2 < t \le t_f, \end{cases}
$$

$$
u^*(t) = \begin{cases} A_{\max} + dv_l(t) + c)/b, & 0 \le t \le t_1, \\ u_k(t), & t_1 < t \le t_2, \\ (A_{\min} + dv_r(t) + c)/b, & t_2 < t \le t_f, \end{cases}
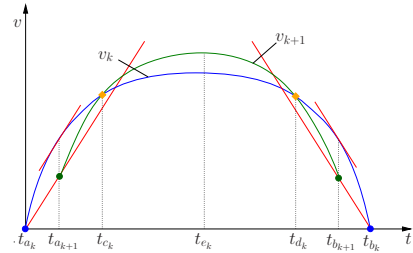$$

Fig. 3. Update scheme of switch times.

Theorem 3.7: The sequences $\{t_{a_k}\}$ and $\{t_{b_k}\}$ as updated in Algorithm 1 converge monotonically to optimal switch times t_1^* and t_2^*, respectively, as $k \to \infty$.

C. Solve Problem 1

Earlier work [15] has shown that an optimal solution of Problem 1 may contain as many as five phases and the structure is as follows: an acceleration constrained arc in the
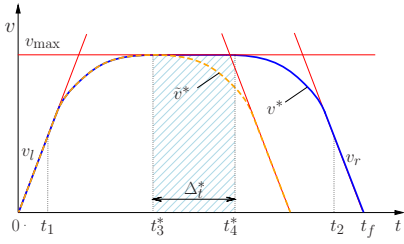
Fig. 4. Optimal speed trajectory with speed constraint.

first phase, an unconstrained arc in the second phase, an speed constrained arc in the third phase, another unconstrained arc in the forth phase, and an deceleration constrained arc in the fifth phase. We consider the general case, i.e., an optimal solution consists of all five phases. Given the structure of an optimal solution, one can always formulate an MBVP and turn to NLP solvers. Solving the MBVP corresponding to Problem 1 directly is more challenging than the ACOCP case due to increased number of variables. Further analysis of structures of optimal solutions for Problems 3 and 1 enables us to recognize that the key to solving Problem 1 is to determine the time interval when the speed constraint is active. In this section, we propose Algorithm 2 to deal with the speed constraint, i.e., determine the time interval of the speed constraint arc. Algorithm 2 relies on a partial equivalence between an optimal solution to Problem 1 and an optimal solution to a particular ACOCP, which can be solved using Algorithm 1. The partial equivalence is established by Theorem 3.8, the proof of which is given in Appendix.

Theorem 3.8: Let (x^*, v^*, u^*) be the optimal solution to Problem 1 with final position x_f and final time t_f. Suppose the speed constraint $v \leq v_{\max}$ is active on the interval $[t_3^*, t_4^*]$, where t_3^* and t_4^* are the optimal switch times at which v^* enters and exits the speed constrained arc. Let $\Delta_t^* = t_4^* - t_3^*$, and let $(\tilde{x}^*, \tilde{v}^*, \tilde{u}^*)$ be the optimal solution to Problem 1 without the speed constraint and with the final position $x_f - \Delta_t^* v_{\max}$ and the final time $t_f - \Delta_t^*$, then (x^*, v^*, u^*) and $(\tilde{x}^*, \tilde{v}^*, \tilde{u}^*)$ are related by

$$\begin{cases} x^*(t) = \tilde{x}^*(t) \\ v^*(t) = \tilde{v}^*(t) \quad, \quad t \in [0, t_3^*], \\ u^*(t) = \tilde{u}^*(t) \end{cases} \quad (20)$$

$$\begin{cases} x^*(t) = \tilde{x}^*(t - \Delta_t^*) + \Delta_t^* v_{\max} \\ v^*(t) = \tilde{v}^*(t - \Delta_t^*) \quad\quad , t \in [t_4^*, t_f]. \quad (21) \\ u^*(t) = \tilde{u}^*(t - \Delta_t^*) \end{cases}$$

Algorithm 2 is essentially a bisection algorithm combined with the Newton's update to find the root to the equation $\eta(\delta) = 0$. The Newton's update can improve the convergence rate when the current update is close to the root of $\eta(\delta) = 0$. We have the following theorem about the convergence of Algorithm 2. The proof of Theorem 3.9 is given in Appendix.

Theorem 3.9: Algorithm 2 converges to an optimal solution of Problem 1.

IV. VALIDATION OF ALGORITHMS

The computation speed of Algorithm 2 and its capability to deal with constraints are evaluated in this section. Algo-

---

Algorithm 2 Algorithm for Problem 1

1: $i \Leftarrow 1$, $\delta_i \Leftarrow 0$, $\eta_i \Leftarrow 1$, and choose a small tolerance parameter $0 < \varepsilon \ll 1$
2: **while** $|\eta_i| > \varepsilon$ **do**
3:    apply Algorithm 1 to solve the acceleration constrained problem on interval $[0, t_f - \delta_i]$ with BC $v(0) = 0$, $x(0) = 0$, $v(t_f - \delta_i) = 0$, $x(t_f - \delta_i) = x_f - v_{\max}\delta_i$. Denote the solution by (x_i, v_i, u_i).
4:    solve $\dot{v}_i(t_{s_i}) = 0$ for t_{s_i}
5:    $\eta_i \Leftarrow v(t_{s_i}) - v_{\max}$
6:    **if** i=1 **then**
7:      **if** $\eta_i \leq 0$ **then**
8:        $\eta_i \Leftarrow 0$
9:      **else**
10:        $\delta_L \Leftarrow 0$, $\delta_U \Leftarrow \Delta_w = \frac{x_f}{v_{\max}} + \frac{v_{\max}}{2A_{\min}} - \frac{v_{\max}}{2A_{\max}}$
11:        $\delta_{i+1} \Leftarrow \frac{1}{2}(\delta_L + \delta_U)$
12:      **end if**
13:    **else**
14:      **if** $\eta_i < 0$ **then**
15:        $\delta_U \Leftarrow \delta_i$
16:      **else**
17:        $\delta_L \Leftarrow \delta_i$
18:      **end if**
19:      $\delta_{i+1} \Leftarrow \delta_i - \left(\frac{\eta_i - \eta_{i-1}}{\delta_i - \delta_{i-1}}\right)^{-1} \eta_i$
20:      **if** $\delta_{i+1} > \delta_U$ or $\delta_{i+1} < \delta_L$ **then**
21:        $\delta_{i+1} = \frac{1}{2}(\delta_L + \delta_U)$
22:      **end if**
23:    **end if**
24:    $i \Leftarrow i + 1$
25: **end while**
26: $t_s \Leftarrow t_{s_{i-1}}$, $\Delta_t^* \Leftarrow \delta_i$
27: $(\tilde{x}^*, \tilde{v}^*, \tilde{u}^*) \Leftarrow (x, v, u)$
28: **return** x^*, v^*, u^*, where

$$x^*(t) = \begin{cases} \tilde{x}^*(t), & 0 \leq t \leq t_s, \\ \tilde{x}^*(t_s) + v_{\max}(t - t_s), & t_s < t \leq t_s + \Delta_t^*, \\ \tilde{x}^*(t - \Delta_t^*) + v_{\max}\Delta_t^*, & t_s + \Delta_t^* < t \leq t_f, \end{cases}$$

$$v^*(t) = \begin{cases} \tilde{v}^*(t), & 0 \leq t \leq t_s, \\ v_{\max}, & t_s < t \leq t_s + \Delta_t^*, \\ \tilde{v}^*(t - \Delta_t^*), & t_s + \Delta_t^* < t \leq t_f, \end{cases}$$

$$u^*(t) = \begin{cases} \tilde{u}^*(t), & 0 \leq t \leq t_1, \\ (dv_{\max} + c)/b, & t_1 < t \leq t_2, \\ \tilde{u}^*(t - \Delta_t^*), & t_2 < t \leq t_f. \end{cases}$$

---

rithms 1-2 are implemented in Matlab®, and tested on a desktop computer with an Intel® Core™2 2.4GHz processor. Algorithm 2, with Algorithm 1 embedded, is used to compute energy-optimal trajectories for 64 positioning tasks where $x_f, t_f, A_{\max}, A_{\min}, d_0, c_0, J$ are different.

A. Computation Speed

For Algorithm 2, among all test cases, the shortest computation time for a single task is 3.1msec, which corresponds to solving the LUOCP; the longest computation time is 33.7msec for a task involving active acceleration and speed constraints; the average computation time of all 64 cases is 6.9msec. One

Fig. 5. Trajectories (x, v, u) for task #1.
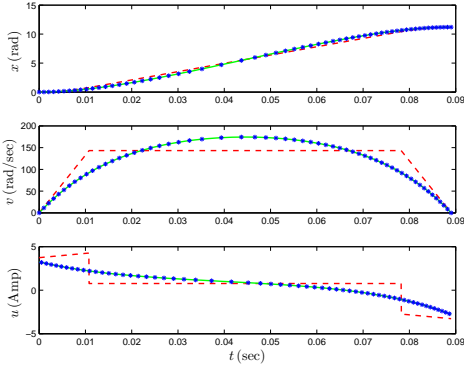
Fig. 6. Trajectories (x, v, u) for task #2.

can argue that Algorithm 2 is suitable for real-time energy-optimal trajectory generation. As a comparison, Problem 1 is also solved via numerical optimization methods including the mesh refinement method in [26]. Numerical optimization-based algorithms typically take 2.5-4.8sec to solve a single task with acceptable accuracy. For all cases, Algorithm 2 and various numerical optimization algorithms produce the same energy-optimal trajectories. This confirms that Algorithm 2 indeed solves Problem 1. For illustration purposes, Figures 5-7 give trajectories generated by using different methods for 3 test cases. Particularly, the green solid and the blue star represent energy-optimal trajectories computed using Algorithm 2 and the mesh refinement method, respectively. The red dash depicts trajectories generated by the trapezoidal method. The main advantage of Algorithm 2, as compared to numerical optimization approaches, is its low computational burden.

B. Satisfaction of Constraints

It is not difficult to verify Algorithm 2 produces trajectories satisfying all constraints for all cases. Figures 5-7 present trapezoidal and energy-optimal trajectories for three example tasks. To simplify the presentation, let the final time be parameterized by $t_f = T_0(1 + \alpha)$, where T_0 is constant and α is a relaxation factor. The three example tasks share the following problem data

$$A_{\max} = 13260\text{rad/sec}^2, \quad A_{\min} = -13260\text{rad/sec}^2,$$
$$V_{\max} = 314.16\text{rad/sec}, \quad J = 7.2 \times 10^{-5}\text{kgm}^2,$$
$$R = 5.06\text{Ohm}, \quad K_t = 0.2723\text{Nm/A},$$
$$c_0 = 6.37 \times 10^{-1}\text{N}, \quad d_0 = 1.01 \times 10^{-3}\text{Nsec/m},$$

and have the following different problem data

task #1: $T_0 = 0.0592$sec, $x_f = 11.2$rad, $\alpha = 0.5$,
task #2: $T_0 = 0.0592$sec, $x_f = 11.2$rad, $\alpha = 0.1$,
task #3: $T_0 = 0.166$sec, $x_f = 44.7$rad, $\alpha = 0.05$.

Optimal trajectories shown in Figures 5-7 exhibit different structures. For task #1, the energy-optimal trajectory merely consists of an unconstrained arc. For task #2, the energy-optimal trajectory does not have speed constrained arcs. For task #3, the energy-optimal trajectory contains five phases:
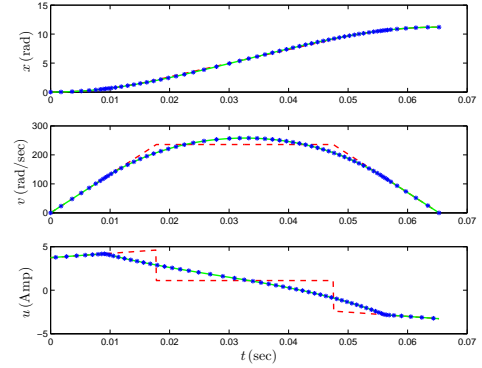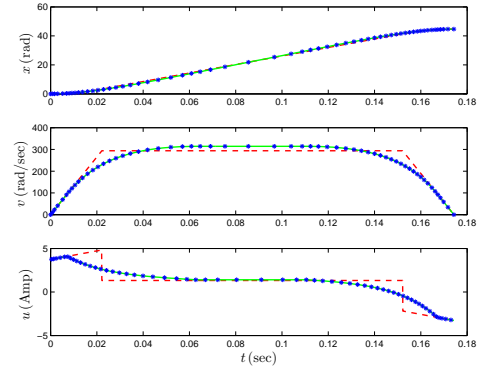
Fig. 7. Trajectories (x, v, u) for task #3.

acceleration constrained arc, unconstrained arc, speed constrained arc, another unconstrained arc, and deceleration constrained arc.

V. EXPERIMENTAL VALIDATION

Experiments are conducted to compare energy consumptions of the following two systems: the energy-optimal system which uses energy-optimal trajectories as references, and the conventional system which uses trapezoidal speed trajectories as references. Experimental results verify that the energy-optimal system achieves comparable tracking performance but consumes less energy than the conventional system.

A. The Testbed

The testbed comprises a Mitsubishi Electric's AC servomotor HF-MP43K with a flexible inertia load, an amplifier MR-J3-40A1, dSPACE® ACE Kit 1104, and Matlab/Simulink®. The testbed is illustrated by Fig. 8 where the black solid and the red dash represent signal and power flows, respectively. The input signal of the testbed is the position trajectory $x^*(t)$ which is generated offline using either Algorithm 2 or the trapezoidal method according to a positioning task data. The position trajectory $x^*(t)$ is fed through a low pass filter to eliminate high frequency components and produce the reference position, speed and acceleration trajectories denoted by x_r, v_r, and a_r, respectively; the tracking controller produces the torque reference $u_r(t)$, linearly proportional to current,
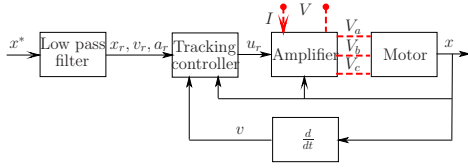
Fig. 8. Architecture of the testbed.

Fig. 9. Difference between (x^*, v^*) and (x_r, v_r), task #3.

based on the references x_r, v_r, a_r and the measured position signal x; the amplifier MR-J3-40A1 runs in the torque control mode and drives the servomotor by generating three-phase voltages V_a, V_b, V_c according to $u_r(t)$. The amplifier acts as the power supply of itself and the motor, and the torque controller. Since the amplifier runs in the torque control mode, the testbed including the amplifier and the motor is consistent with the model (1).

The dSPACE executes the data acquisition and real-time tracking control tasks. For data acquisition, the dSPACE collects three signals: the motor position x, the voltage V and current I. The motor position is sensed by the encoder and the measured signal x enters the dSPACE through the Digital Incremental Encoder Interface (DIEI); the voltage $V(t)$ and current $I(t)$ are measured by Fluke DP120 and Fluke i30s, respectively. Both V and I are fed into the dSPACE through Analog Input (AI) ports. The control signal u_r drives the amplifier through an Analog Output (AO) port. While performing a real-time positioning task, the dSPACE operates at a sampling frequency of 10kHz, i.e., all DIEI/AI/AO signals are sampled every 100 μs. Note that the position signal x to the DIEI is relayed by the amplifier; the interface between the amplifier and the DIEI is compatible because the DIEI expects its inputs from incremental encoders, and the amplifier only outputs incremental encoder signals in the torque control mode. The amplifier runs in the setting that the resolution of the relayed signal x is 12500 pulses per revolution.

B. The Low Pass Filter and Tracking Controller

A third order low pass filter, smoothing $x^*(t)$, is given by

$$\dot{\xi}_1 = \xi_2,$$
$$\dot{\xi}_2 = \xi_3,$$
$$\dot{\xi}_3 = -\lambda^3 \xi_1 - 3\lambda^2 \xi_2 - 3\lambda \xi_3 + \lambda^3 x^*(t),$$
$$y = \xi,$$

where $\xi = (\xi_1, \xi_2, \xi_3)^T$, $y = (x_r, v_r, a_r)^T$, and $\lambda = 5000$. The low pass filter implicitly imposes jerk constraints on reference trajectories which are easier to track. Since all poles of the low pass filter, located at -5000, are much larger than the bandwidth of the servomotor system (130rad/sec), the phase shift of the reference trajectory introduced by the low pass filter is negligible. Fig. 9 shows differences between inputs and outputs of the low pass filter for task #3 for both the energy-optimal and trapezoidal cases. The amplitude of $x^* - x_r$ is less than 0.25rad, or 0.56% of x_f – thus negligible. The amplitude of $v^* - v_r$ is relatively large, which is not critical since the position tracking performance is of interest.
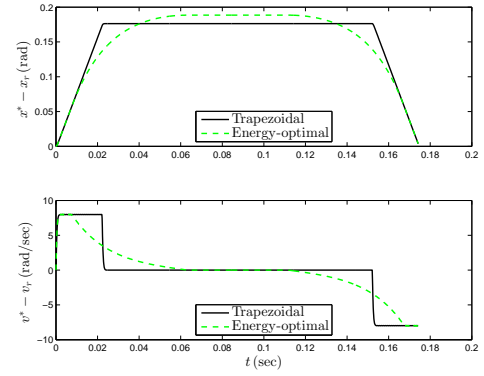
The tracking controller consists of a feed-forward u_{ff} and a feedback u_{fb}. The u_{ff} is based on the dynamics (1) and takes the following form

$$u_{ff} = \frac{1}{b} \left(k_{fa} a_r + k_{fd} dv_r + k_{fc} c \right), \qquad (22)$$

where k_{fa}, k_{fd}, k_{fc} are constants. The u_{fb} is a PD control and given by

$$u_{fb} = k_x(x_r - x) + k_v(v_r - v), \qquad (23)$$

where k_x, k_v are constant, and v is the speed signal obtained by differentiating x. All constants $k_{fa}, k_{fd}, k_{fc}, k_x$, and k_v are obtained by trial and error to achieve satisfactory position tracking performance.

C. Experimental Results

Experiment is performed to validate that the energy-optimal system can achieve comparable position tracking performance as the trapezoidal system but consume less energy. During the validation, both the energy-optimal and trapezoidal systems complete 18 positioning tasks parameterized by (x_f, α) with $x_f \in \mathcal{F} = \{1.86, 11.2, 44.7\}$, and $\alpha \in \mathcal{T} = \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$. Since all tasks lead to similar qualitative conclusion, we take the positioning task #3 as an example and look into detailed experimental results. Task #3 has been used in Section IV to verify the proposed algorithm, and the corresponding trajectory generation results are shown in Fig. 7. In lining up with experiment, concise simulation is conducted for verification. Simulation diagram is given in Fig. 10, where the tracking control is given by (22)-(23), the motor model is given by (1), and e is to mimic the measurement noise. The simulation diagram is slightly different from Fig. 8 since the low pass filter and the amplifier blocks are not simulated. Parameters of the tracking controller take different values during simulation and experiment. Due to the perfect modeling assumption, almost perfect position and velocity tracking performance is achieved during simulation. That is: position and velocity tracking errors are almost zero. Hereinafter, reference trajectories are thus omitted in figures to avoid redundancy.
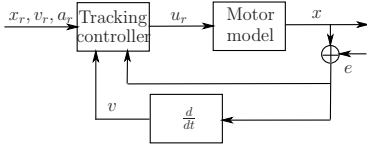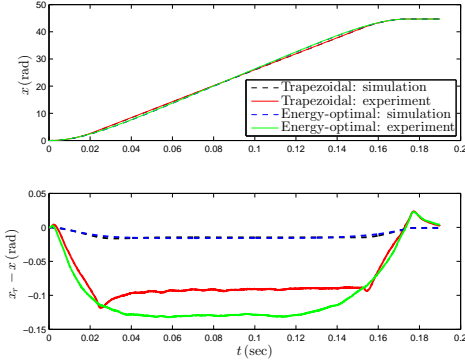
Fig. 10. Simulation diagram of the testbed.

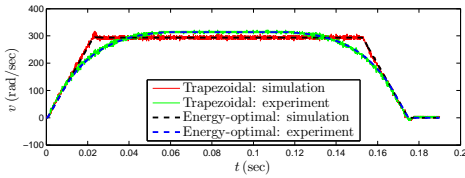Fig. 11. Position trajectories for task #3.

Fig. 12. Speed trajectories for task #3.

1) Position Tracking Performance: Fig. 11 gives the simulation and experimental position trajectories for task #3. According to Fig. 11, the energy-optimal and trapezoidal position trajectories are distinct from each other but the difference is not remarkable. Given the reference x_r and the position trajectories from simulation and experiments, Fig. 11 also shows the position tracking errors for the trapezoidal and the energy-optimal cases. Since experimental position tracking errors for both cases are within 0.15rad, both systems have similar position tracking performance.

2) Velocity Tracking Performance: Fig. 12 shows the speed trajectories v for task #3, where, for simplicity, v are obtained by differentiating x. More elegant solutions to infer unmeasured states from measured outputs in practice are available e.g. Kalman filters and Luenberger observers. In Fig. 12, the energy-optimal and trapezoidal speed trajectories are remarkably different from each other, and v for both cases can roughly track their individual reference v_r. Fig. 13 plots the speed tracking errors. Both the energy-optimal and trapezoidal systems yield similar speed tracking errors: 20rad/sec.

3) Energy Efficiency: For each positioning task, the energy consumptions of the entire testbed for both the energy-optimal and trapezoidal cases are computed as $E(t) = \int_0^t V(\tau)I(\tau)\mathrm{d}\tau$, where t is the time when the testbed settles. For task #3, experiment shows that the energy-optimal system consumes 11.32% less energy than the trapezoidal case. For all 18 positioning tasks, energy consumption comparison results are summarized
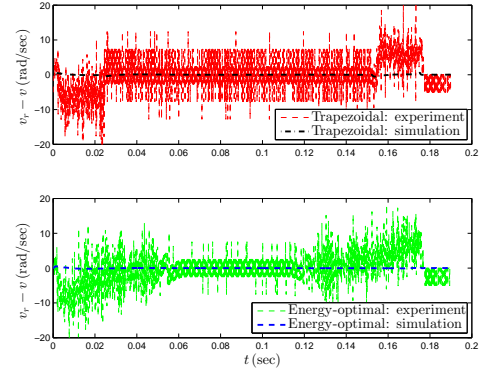

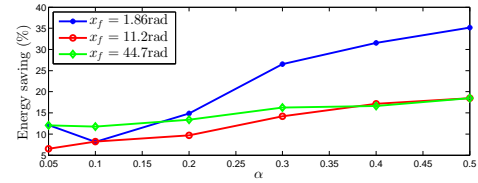Fig. 13. Speed tracking errors for task #3.

Fig. 14. Energy efficiency.

in Fig. 14, where the horizontal axis is the relaxation factor α, and the vertical axis denotes energy saving percentages of the energy-optimal system versus the trapezoidal case. As shown in Fig. 14, the energy-saving percentages vary case by case. For all 18 cases, the energy-optimal system consumes 16.2% less energy than the conventional system. The energy saving percentage in experiment are however slightly off that predicted by simulation. The discrepancy is due to factors including uncertainties of the system and energy consumption models. Although the energy efficiency varies case by case, experimental results however corroborate consistently that the energy-optimal system achieves similar tracking performance but consumes less energy than the conventional system.

4) Design Tradeoff: For positioning systems, the tracking accuracy of the final position is crucial. The position tracking error during transient is also important to ensure certain properties, for instance satisfaction of all constraints, optimality, etc. The tracking performance of the testbed system is influenced by references, measurement noises, motor dynamics, the tracking controller, etc. There exists a well-known design tradeoff between fast tracking and robustness to measurement noises. In experiments, we weight the tracking performance more than robustness, which necessarily results in relatively large PD gains. Accordingly, the resultant testbed is likely sensitive to measurement noises.

5) Harmonics in v: Although harmonics analysis can be carried out systematically, we merely give brief discussions on the large harmonics in v, which are given by Figures 12-13. This is because first, the presence of harmonics does not affect conclusions, and second, the harmonics analysis is not the main focus of this paper. To find out the factors inducing large harmonics in the speed v, we examine the experimental speed tracking error as shown in Fig. 15 where the solid black and blue lines represent the signals obtained by feeding the
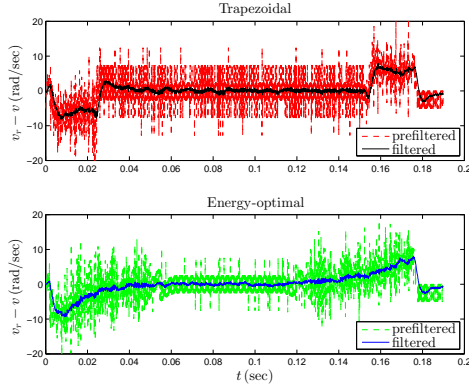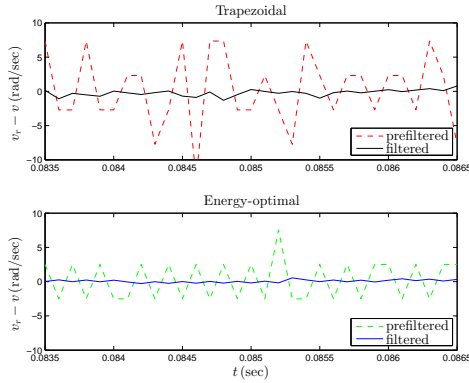
Fig. 15. Speed tracking errors for task #3.

Fig. 16. Zoomed speed tracking errors for task #3.

speed tracking errors $v_r - v$ through a first-order low pass filter: $\frac{1000}{s+1000}$.

Since the filtered speed error signals are quite smooth, one can see that the harmonics lies in the high frequency range. Fig. 16, obtained by zooming in Fig. 15 , shows that the harmonic frequencies are half of the sampling frequency. It is a reasonable conjecture that the quantization error in the measured position x might be the main factor for the harmonics. Indeed, all speed v are obtained by differentiating the measured position x, which apparently amplifies the measurement noise including quantization errors. Considering that the sample frequency is 10Khz, the measurement noise at the frequency 5KHz could be amplified $\pi \times 10^4$ times. Assuming the quantization error is ± 0.5 pulse in the position measurement, with the encoder resolution 12500 pulses per revolution, the induced speed error is ± 7.90rad/sec. This calculation result is roughly consistent with the amplitudes of curves shown in Figures 13 and 16.

The aforementioned analysis can also be validated by simulation, where a pulse signal is injected into the measurement x. The pulse signal $e(t)$ has an amplitude of $2\pi/12500$rad, a frequency of 5kHz, and its pulse width is 50% of the period. Simulation results are summarized by Fig. 17, where the induced harmonics in v has an amplitude close to 6rad/sec. One can therefore conclude that the measurement noise including quantization error in the position measurement, and the differentiation of x in order to obtain the speed, contribute
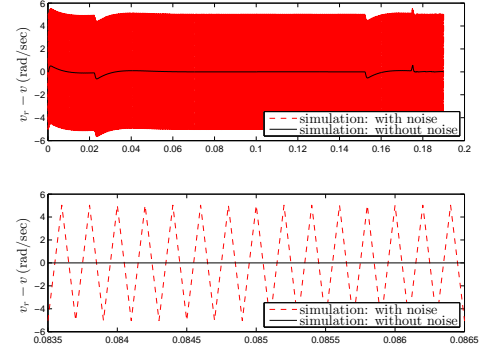

Fig. 17. Speed tracking errors for task #3, trapezoidal case.

to the large harmonics in speed trajectories. It is understood that the large harmonics can be presumably suppressed by choosing a lower sample frequency, for instance 1Khz, increasing the encoder resolution, and designing sophisticated speed estimators instead of a differentiator to obtain v, etc.

VI. CONCLUSION

This paper proposed computationally efficient algorithms to generate energy-optimal trajectories for a servomotor system subject to acceleration and speed constraints. Instead of solving the MBVP directly, the linear constrained optimal control problem (LCOCP) was transformed to determine the optimal time interval of the speed constrained arc by solving a sequence of acceleration constrained optimal control problems (ACOCPs). Each ACOCP is parameterized by a specific choice of the time interval of the speed constrained arc and equivalent to the LCOCP. The ACOCP was further reduced to determine optimal switch times of acceleration constrained arcs by solving a sequence of two-point boundary value problems (TBVPs), whose BC is uniquely defined by the switch times. It was proved that the proposed algorithm is guaranteed to converge to an optimal solution, and incurs low computational burden as compared to various numerical optimization methods. Experiments were performed to verify that the servomotor system using an energy-optimal trajectory as reference indeed leads to less energy consumption.

APPENDIX

A. Definitions of M and G

With $\Delta_t = t_f - t_0$,

$$M(t_f, t_0) = \begin{bmatrix} M_{1,1} & M_{1,2} & M_{1,3} \\ M_{2,1} & M_{2,2} & M_{2,3} \\ M_{3,1} & M_{3,2} & M_{3,3} \end{bmatrix}$$

$$M_{1,1} = 1;$$

$$M_{1,2} = \frac{4}{p^2}\left[\frac{p}{2}\sinh\left(\frac{p\Delta_t}{2}\right) + A_1\cosh\left(\frac{p\Delta_t}{2}\right) - A_1\right];$$

$$M_{1,3} = \frac{4A_2}{p^2}\left(\cosh\left(\frac{p\Delta_t}{2}\right) - 1\right);$$

$$M_{2,2} = \cosh\left(\frac{p\Delta_t}{2}\right) + \frac{2A_1}{p}\sinh\left(\frac{p\Delta_t}{2}\right);$$

$$M_{2,3} = \frac{2A_2}{p} \sinh\left(\frac{p\Delta_t}{2}\right)$$

$$M_{3,1} = 0;$$

$$M_{3,2} = \frac{2A_3}{p} \sinh\left(\frac{p\Delta_t}{2}\right);$$

$$M_{3,3} = \cosh\left(\frac{p\Delta_t}{2}\right) - \frac{2A_1}{p} \sinh\left(\frac{p\Delta_t}{2}\right)$$

$$G_{1,1} = \Delta_t;$$

$$G_{1,2} = \frac{4}{p^2} [M_{2,2} - 1 - A_1\Delta_t];$$

$$G_{1,3} = \frac{8A_2}{p^3} \sinh\left(\frac{p\Delta_t}{2}\right) - \frac{4A_2}{p^2}\Delta_t;$$

$$G_{2,1} = 0;$$

$$G_{2,2} = \frac{4A_1}{p^2}\left(\cosh\left(\frac{p\Delta_t}{2}\right) - 1\right) + \frac{2}{p} \sinh\left(\frac{p\Delta_t}{2}\right)$$

$$G_{2,3} = M_{1,3}$$

$$G_{3,1} = 0;$$

$$G_{3,2} = \frac{A_3}{A_2} M_{1,3}$$

$$G_{3,3} = -\frac{4A_1}{p^2}\left(\cosh\left(\frac{p\Delta_t}{2}\right) - 1\right) + \frac{2}{p} \sinh\left(\frac{p\Delta_t}{2}\right)$$

B. Proof of Theorem 3.7

Proof: Suppose that the acceleration constraints (2b) are active in the optimal solution to the ACOCP. We first show that Algorithm 1 ensures $t_{a_{k+1}} \geq t_{a_k}$ and $t_{b_{k+1}} \leq t_{b_k}$. Let (x_k, v_k, u_k) be the optimal solution at the kth step with BC $x_k(t_{a_k}) = x_l(t_{a_k})$, $v_k(t_{a_k}) = v_l(t_{a_k})$, $x_k(t_{b_k}) = x_l(t_{b_k})$, $v_k(t_{b_k}) = v_l(t_{b_k})$. When $\dot{v}_{a_k} \leq A_{\max}$, $t_{a_{k+1}} \geq t_{a_k}$ holds since $t_{a_{k+1}} = t_{a_k}$ by Algorithm 1. Therefore, it suffices to consider only the case that $\dot{v}_k(t_{a_k}) > A_{\max}$, and $\dot{v}_k(t_{b_k}) < A_{\min}$.

Because $\dot{v}_k(t_{a_k}) > A_{\max} > 0$, $\dot{v}_k(t_{b_k}) < A_{\min} < 0$, and $\dot{v}$ is continuous, both $\dot{v}_k(\tau_l) = A_{\max}$ and $\dot{v}_k(\tau_r) = A_{\min}$ have at least one root on (t_{a_k}, t_{b_k}). Because $\dot{v}_k(t_{a_k})\dot{v}_k(t_{b_k}) < 0$, $\dot{v}_k$ is strictly monotone according to Proposition 3.2. This implies that both $\dot{v}_k(\tau_l) = A_{\max}$ and $\dot{v}_k(\tau_r) = A_{\min}$ have a unique solution, which can only be on (t_{a_k}, t_{b_k}). Therefore, we must have $t_{a_{k+1}} > t_{a_k}$ and $t_{b_{k+1}} < t_{b_k}$.

Next we show that $\dot{v}_{a_{k+1}} > A_{\max}$, and $\dot{v}_{b_{k+1}} < A_{\min}$, which ensure that the same conditions in the current iteration still holds in the next iteration. Since $v_k(t_{a_k}) = v_l(t_{a_k})$, and $\dot{v}_k(t) > A_{\max} = \dot{v}_l$ for $t \in [t_{a_k}, t_{a_{k+1}})$, we have $v_k(t_{a_{k+1}}) > v_l(t_{a_{k+1}}) = v_{k+1}(t_{a_{k+1}})$ by the Comparison Lemma. Similarly, we have $v_k(t_{b_{k+1}}) > v_l(t_{b_{k+1}})$.

Because $x_k(t_{a_k}) = x_l(t_{a_k})$ and $v_k(t) > v_l(t)$ for $t \in (t_{a_k}, t_{a_{k+1}})$, we have $x_k(t_{a_{k+1}}) > x_l(t_{a_{k+1}})$ by the Comparison Lemma. Similarly, we have $x_k(t_{b_{k+1}}) < x_r(t_{b_{k+1}})$. Therefore, the following is true,

$$x_k(t_{b_{k+1}}) - x_k(t_{a_{k+1}}) < x_r(t_{b_{k+1}}) - x_l(t_{a_{k+1}})$$
$$= x_{k+1}(t_{b_{k+1}}) - x_{k+1}(t_{a_{k+1}}), \quad (24)$$

which implies that the average of v_k on $[t_{a_{k+1}}, t_{b_{k+1}}]$ is smaller than that of v_{k+1}, which further implies that there exists

$\tau_k \in (t_{a_{k+1}}, t_{b_{k+1}})$ such that $v_k(\tau_k) < v_{k+1}(\tau_k)$. Since v_k and v_{k+1} are continuous, and we have shown that $v_k(t_{a_{k+1}}) > v_{k+1}(t_{a_{k+1}})$ and $v_k(t_{b_{k+1}}) > v_{k+1}(t_{b_{k+1}})$, v_k and v_{k+1} must intersect at two points on $[t_{a_{k+1}}, t_{b_{k+1}}]$ following the first part of Proposition 3.1. Without loss of generality, denote these two points by t_{c_k} and t_{d_k} with $t_{a_{k+1}} < t_{c_k} < \tau_k < t_{d_k} < t_{b_{k+1}}$.

Because $v_k(t_{c_k}) - v_{k+1}(t_{c_k}) = 0$, $v_k(t_{d_k}) - v_{k+1}(t_{d_k}) = 0$, there must exists $t_{e_k} \in (t_{c_k}, t_{d_k})$ such that $\dot{v}_k(t_{e_k}) - \dot{v}_{k+1}(t_{e_k}) = 0$ following the Mean Value Theorem. According to the second part of Proposition 3.1, t_{e_k} is the only real solution of $\dot{v}_k(t_{e_k}) - \dot{v}_{k+1}(t_{e_k}) = 0$, therefore we have $\dot{v}_k(t) < \dot{v}_{k+1}(t)$ for any $t < t_{e_k}$ and $\dot{v}_k(t) > \dot{v}_{k+1}(t)$ for any $t > t_{e_k}$. Specifically, at $t_{a_{k+1}}$ and $t_{b_{k+1}}$, we have $A_{\max} = \dot{v}_k(t_{a_{k+1}}) < \dot{v}_{k+1}(t_{a_{k+1}})$ and $A_{\min} = \dot{v}_k(t_{b_{k+1}}) > \dot{v}_{k+1}(t_{b_{k+1}})$, which guarantees that the same condition in the kth step still holds for the $(k+1)^{\text{th}}$ step.

Algorithm 1 generates two monotone sequences $\{t_{a_k}\}$ and $\{t_{b_k}\}$ with $t_{a_k} \leq t_{a_{k+1}}$ and $t_{b_k} \geq t_{b_{k+1}}$ for any $k \in \mathbb{N}$, $k > 1$. Since $\{t_{a_k}\}$ and $\{t_{b_k}\}$ are bounded from both below and above, they must converge monotonically as $k \to \infty$.

Let $t_1 = \lim_{k \to \infty} t_{a_k}$ and $t_2 = \lim_{k \to \infty} t_{b_k}$. Since t_1, t_2 are the fixed-point of Algorithm 1, we must have $\dot{v}(t_1) = A_{\max}$ and $\dot{v}(t_2) = A_{\min}$. Then the solution with switch time t_1 and t_2 satisfies all NOCs, hence, is optimal.

If only the acceleration constraint is active at the beginning of the optimal solution, $t_{b_k} = t_f$ for all $k \in \mathbb{N}$, and it can be shown similarly that $\{t_{a_k}\}$ is a monotone increasing sequence. A similar conclusion holds when only the deceleration constraint is active. In these two cases, Algorithm 1 can still identify the optimal switch time. The proof is complete. ∎

C. Proof of Theorem 3.8

Proof: On the speed constrained arc, we have $u^*(t) = (dv_{\max} + c)/b$, $t \in [t_3^*, t_4^*]$. Without loss of generality, we may assume that the acceleration constraints are active on $[0, t_1^*]$ and $[t_2^*, t_f]$ with switch times t_1^* and t_2^*. From (6), the costate λ_v on the unconstrained arcs is given by

$$\lambda_v(t) = -2Ru^*(t) - K_t v^*(t), \quad t \in [t_1^*, t_3^*] \cup [t_4^*, t_2^*].$$

Since u^*, x^* and v^* are continuous with respect to t, we have $\lambda_v^*(t_3^{*-}) = \lambda_v^*(t_4^*)$. According to (10), the optimal solution on $[t_1^*, t_3^*)$ is given by

$$X^*(t) = M(t, t_1^*)X(t_1^*) + G(t, t_1^*)B, \ t \in [t_1^*, t_3^*), \quad (25)$$

and $X^*(t_3^*) = M(t_3^*, t_1^*)X^*(t_1^*) + G(t_3^*, t_1^*)B$. On $[t_4^*, t_2^*]$, the optimal solution is given by

$$X^*(t) = M(t, t_4^*)X^*(t_4^*) + G(t, t_4^*)B, \quad t \in [t_4^*, t_2^*]. \quad (26)$$

With a change of the time variable, (26) can be written as

$$X^*(t + \Delta_t^*) = M(t + \Delta_t^*, t_4^*)X^*(t_4^*) + G(t + \Delta_t^*, t_4^*)B$$
$$= M(t, t_3^*)\begin{bmatrix} x^*(t_3^*) + \Delta_t^* v_{\max} \\ v^*(t_3^*) \\ \lambda_v^*(t_3^{*-}) \end{bmatrix} + G(t, t_3^*)B, \ t \in [t_3^*, t_2^* - \Delta_t^*].$$

Since $M_{1,1} = 1$, $M_{2,1} = M_{3,1} = 0$, we have

$$\begin{bmatrix} x^*(t + \Delta_t^*) - \Delta_t^* v_{\max} \\ v^*(t + \Delta_t^*) \\ \lambda_v^*(t + \Delta_t^*) \end{bmatrix} = M(t, t_3^*)\begin{bmatrix} x^*(t_3^*) \\ v^*(t_3^*) \\ \lambda_v^*(t_3^{*-}) \end{bmatrix} + G(t, t_3^*)B$$

$$= M(t, t_3^*)\left(M(t_3^*, t_1^*)X^*(t_1^*) + G(t_3^*, t_1^*)B\right) + G(t, t_3^*)B$$
$$= M(t, t_1^*)X^*(t_1^*) + G(t, t_1^*)B, \quad t \in [t_3^*, t_2^* - \Delta_t^*]. \quad (27)$$

Because the right-hand-side of (27) is identical to that of (25), one can see that for $t \in [t_3^*, t_2^* - \Delta_t^*]$,

$$(x^*(t + \Delta_t^*) - \Delta_t^* v_{\max}, v^*(t + \Delta_t^*), \lambda_v^*(t + \Delta_t^*))$$

is an extension of the optimal solution (x^*, v^*, λ_v^*) from $[t_1^*, t_3^*)$ to $[t_3^*, t_2^* - \Delta_t^*]$.

Let $(\tilde{x}^*(t), \tilde{v}^*(t), \tilde{u}^*(t)) = (x^*(t), v^*(t), u^*(t))$ for $t \in [t_1^*, t_3^*)$, and $(\tilde{x}^*(t), \tilde{v}^*(t), \tilde{u}^*(t)) = (x^*(t + \Delta_t^*) - \Delta_t^* v_{\max}, v^*(t + \Delta_t^*), u^*(t + \Delta_t^*))$ for $t \in [t_3^*, t_2^* - \Delta_t^*]$, then $(\tilde{x}^*(t), \tilde{v}^*(t), \tilde{u}^*(t))$ is the optimal solution to the unconstrained problem with initial condition $\tilde{x}^*(t_1^*) = x_l(t_1^*)$, $\tilde{v}^*(t_1^*) = v_l(t_1^*)$ at the initial time t_1^*, and final condition $\tilde{x}^*(t_2^* - \Delta_t^*) = x_r(t_2^*)$, $\tilde{v}^*(t_2^* - \Delta_t^*) = v_r(t_2^*)$ at the final time $t_2^* - \Delta_t^*$.

Extending the definition of $(\tilde{x}^*(t), \tilde{v}^*(t), \tilde{u}^*(t))$ to $[0, t_1^*)$ and $[t_2^* - \Delta_t^*, t_f - \Delta_t^*]$ with

$$\tilde{x}^*(t) = \begin{cases} x_l(t), & t \in [0, t_1^*), \\ x_r(t + \Delta_t^*), & t \in [t_2^* - \Delta_t^*, t_f - \Delta_t^*], \end{cases}$$

$$\tilde{v}^*(t) = \begin{cases} v_l(t), & t \in [0, t_1^*), \\ v_r(t + \Delta_t^*), & t \in [t_2^* - \Delta_t^*, t_f - \Delta_t^*], \end{cases}$$

$$\tilde{u}^*(t) = \begin{cases} u_l(t), & t \in [0, t_1^*), \\ u_r(t + \Delta_t^*), & t \in [t_2^* - \Delta_t^*, t_f - \Delta_t^*], \end{cases}$$

where $u_l(t) = (A_{\max} + dv_l(t) + c)/b$, and $u_r(t) = (A_{\min} + dv_r(t) + c)/b$. Then $(\tilde{x}^*, \tilde{v}^*, \tilde{u}^*)$ is the optimal solution to the acceleration constrained energy-optimal motor control problem with initial condition $x(0) = 0$, $v(0) = 0$ and final condition $x(t_f - \Delta_t^*) = x_f - \Delta_t^* v_{\max}$. $(\tilde{x}^*, \tilde{v}^*, \tilde{u}^*)$ is indeed optimal because it satisfies all NOCs. Note that junctions conditions are satisfied because $\dot{\tilde{v}}^*(t_1^*) = \dot{v}^*(t_1^*) = A_{\max}$ and $\dot{\tilde{v}}^*(t_2^* - \Delta_t^*) = \dot{v}^*(t_2^*) = A_{\min}$. The proof is complete since the relation between (x^*, v^*, u^*) and $(\tilde{x}^*, \tilde{v}^*, \tilde{u}^*)$ as described by (20) is ensured by the construction of $(\tilde{x}^*, \tilde{u}^*)$. ∎

D. Proof of Theorem 3.9

Proof: Consider a trapezoidal speed trajectory defined by

$$w(t) = \begin{cases} A_{\max}t, & 0 \le t \le \frac{v_{\max}}{A_{\max}}, \\ v_{\max}, & \frac{v_{\max}}{A_{\max}} < t \le \frac{v_{\max}}{A_{\max}} + \Delta_w, \\ v_{\max} + A_{\min}t, & \frac{v_{\max}}{A_{\max}} + \Delta_w < t \le t_{f_w}, \end{cases}$$

where Δ_w is defined in Algorithm 2, and $t_{f_w} = v_{\max}/A_{\max} + \Delta_w - v_{\max}/A_{\min}$. Then it can be easily verified that such a trapezoidal speed trajectory is the optimal solution to a minimum-time motor position control problem with the same dynamics, BC, and constraints as those of Problem 1.

Let $\tilde{v}$ denote the energy-optimal solution to the acceleration constrained, but not speed constrained motor position control problem with the final time $t_f - \delta$ and the BC $x(t_f - \delta) = x_f - v_{\max}\delta$. The other unspecified BC is the same as those of Problem 1. Henceforth, we denote this problem as the *relaxed problem*. Also, let t_a and t_b denote the switch times from acceleration constrained arc to unconstrained arc, and from unconstrained arc to deceleration constrained arc. According

to Proposition 3.2, $\dot{\tilde{v}}$ is monotonically decreasing on $[t_a, t_b]$ with $\dot{\tilde{v}}(t_a) > 0$ and $\dot{\tilde{v}}(t_b) < 0$. Therefore, by solving for t_s from $\dot{\tilde{v}}(t_s) = 0$ as in Algorithm 2, we can obtain the maximum value of $\tilde{v}$, which is $\tilde{v}(t_s)$.

Let $\eta(\delta) = \tilde{v}(t_s) - v_{\max}$. Note that $\eta(\delta)$ depends continuously on δ, since $\tilde{v}$ depends continuously on the associated BC, which further depends continuously on δ. Since Algorithm 2 includes a bisection search with updated bounds, in order to prove the convergence of Algorithm 2, we only need to prove that $\eta(0) > 0$, $\eta(\Delta_w) < 0$, and there is a unique $\delta \in (0, \Delta_w)$ such that $\eta(\delta) = 0$.

When $\delta = 0$, the relaxed problem share the same BC and final time with Problem 1 except that the relaxed problem has no speed constraint. If $\eta(0) \le 0$, then $\tilde{v}$ does not violate the speed constraint. Hence, $\tilde{v}$ is also the optimal solution to Problem 1. But $\tilde{v}$ does not contain any speed constrained arc, which is a contradiction. Therefore we must have $\eta(0) > 0$.

In order to show that $\eta(\Delta_w) < 0$, consider a second minimum-time problem with a new final position $x_f - v_{\max}\Delta_w$. Then the second minimum-time problem and the relaxed energy-optimal problem with $\delta = \Delta_w$ share the same final position. It is easily seen that the optimal speed solution $w_2(t)$ to this second minimum-time problem is a triangle, and $\max\{w_2(t)|0 < t < v_{\max}/A_{\max} - v_{\max}/A_{\min}\} = v_{\max}$, as illustrated in Fig. 18. Because for both problems the position increases monotonically with respect to the time, we may parameterize the speed using the position, and write w_2 and $\tilde{v}$ as $w_2(x)$ and $\tilde{v}(x)$. Since w_2 is a minimum-time solution, we must have $w_2(x) \ge \tilde{v}(x)$ for all $x \in [0, x_f - v_{\max}\Delta_w]$. Let x_m be the position at which $w_2(x_m) = v_{\max}$. Suppose, ad absurdum, that there exists $x_s \in [0, x_f - v_{\max}\Delta_w]$ such that $\tilde{v}(x_s) \ge v_{\max}$. Without loss of generality, we may assume that $x_s \in [0, x_m]$. If $x_s \in [0, x_m)$, then $\tilde{v}(x_s) > w_2(x_s)$, which means that $\tilde{v}(x) > \dot{w}_2(x) = A_{\max}$ for some $x \in [0, x_s)$, which is a contradiction. Hence, we can only have $x_s = x_m$. Since in $\tilde{v}$, the acceleration arc and the deceleration constrained arc are connected by an unconstrained arc, $\tilde{v}$ is not constrained by the acceleration constraints in a neighborhood of x_s. It follows that $\tilde{v}' > w_2'$ in $[x_s - \epsilon, x_s + \epsilon]$ for some $\epsilon > 0$, where the prime denotes the derivative with respect to x. However, by integrating $\tilde{v}'$ forward we have $\tilde{v}(x_s + \epsilon) > w_2'(x_s + \epsilon)$, which is a contradiction to the fact that w_2 is the minimum-time solution. Therefore, we must have $\tilde{v}(x) < v_{\max}$ for all $x \in [0, x_f - v_{\max}\Delta_w]$, which is equivalent to $\tilde{v}(t) < v_{\max}$ for all $t \in [0, t_f - \Delta_w]$. Hence we have shown that $\eta(\Delta_w) < 0$.

Suppose there exist two solutions δ_1 and δ_2 to $\eta(\delta) = 0$. We can recover two different solutions to Problem 1 using the two partially equivalent solutions corresponding to δ_1 and δ_2. Theorem 3.8 ensures both recovered solutions are optimal, which contradicts the fact that Problem 1 has a unique solution. Hence, $\eta(\delta) = 0$ has an unique root. ∎

REFERENCES

[1] J. Barraquand, B. Langlois, and J.-C. Latombe, "Numerical potential field techniques for robot path planning," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, no. 2, pp. 224–241, Mar./Apr. 1992.

[2] J. E. Bobrow, S. Dubowsky, and J. S. Gibson, "Time-optimal control of robotic manipulators along specified paths," *Int. J. Robot Res.*, vol. 4, no. 3, pp. 3–17, Sep. 1985.
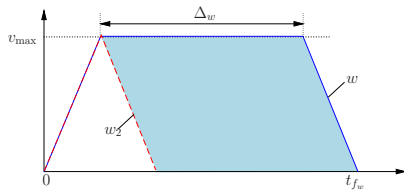
Fig. 18. Two minimum-time solutions.

[3] D. Verscheure, B. Demeulenaere, J. Swevers, J. D. Schutter, and M. Diehl, "Time-optimal path tracking for robots: A convex optimization approach," *IEEE Trans. Automat. Control*, vol. 54, no. 10, pp. 2318–2327, Oct. 2009.

[4] R. Errouissi, M. Quhrouche, W.-H. Chen, and A. M. Trzynadlowski, "Robust nonlinear predictive controller for permanent-magnet synchronous motors with an optimized cost function," *IEEE Trans. Ind. Electr.*, vol. 59, no. 7, pp. 2849–2858, Jul. 2012.

[5] H. K. Khalil, E. G. Strangas, and S. Jurkovic, "Speed observer and reduced nonlinear model for sensorless control of induction motors," *IEEE Trans. Contr. Syst. Technol.*, vol. 17, no. 2, pp. 327–339, 2009.

[6] R. Ortega, N. Barabanov, and G. E. Valderrama, "Direct torque control of induction motors: stability analysis and performance improvement," *IEEE Trans. Automat. Control*, vol. 46, no. 8, pp. 1209–1222, 2001.

[7] T. Geyer, G. Papafotiou, and M. Morari, "Model predictive direct torque control - Part I: concept, algorithm, and analysis," *IEEE Trans. Ind. Electr.*, vol. 56, no. 6, pp. 1894–1905, Jun. 2009.

[8] E. Poirier, M. Ghribi, and A. Kaddouri, "Loss minimization control of induction motor drives based on genetic algorithms," in *Proc. of the IEEE Int. Electric Mach. and Drives Conf.*, Cambridge, MA, Jun. 2001, pp. 475–478.

[9] P. Lambrechts, M. Boerlage, and M. Steinbuch, "Trajectory planning and feedforward design for electromechnical motion systems," *Control Eng. Pract.*, vol. 13, pp. 145–157, Jan. 2005.

[10] M.-H. Park and C.-Y. Won, "Time optimal control for induction motor servo system," *IEEE Trans. Power Electron.*, vol. 6, no. 3, pp. 514–524, Jul. 1991.

[11] C. La-orpacharapan and L. Y. Pao, "Shaped time-optimal feedback control for disk-drive systems with back-electromotive force," *IEEE Trans. Magn.*, vol. 40, no. 1, pp. 85–96, Jan. 2004.

[12] M. Vasak, M. Baotic, I. Petrovic, and N. Peric, "Hybrid theory-based time-optimal control of an electronic throttle," *IEEE Trans. Ind. Electr.*, vol. 54, no. 3, pp. 1483–1494, Jun. 2007.

[13] S. Dodds, "Sliding mode vector control of pmsm drives with minimum energy position following," in *Proc. of the 13th Power Electron. and Motion Control Conf.*, Sep. 2008, pp. 2559–2566.

[14] J. Gregory, A. Olivares, and E. Staffetti, "Energy-optimal trajectory planning for robot manipulators with holonomic constraints," *Syst. Control Lett.*, vol. 61, no. 2, pp. 279–291, Feb. 2012.

[15] Y. Wang, K. Ueda, and S. A. Bortoff, "A Hamiltonian approach to compute an energy efficient trajectory for a servomotor system," *Automatica*, vol. 49, no. 12, pp. 3550–3561, Dec. 2013.

[16] M. H. Korayem, A. Nikoobin, and V. Azimirad, "Maximum load carrying capacity of mobile manipulators: optimal control approach," *Robotica*, vol. 27, no. 1, pp. 147–159, Jan. 2009.

[17] E. S. Sergaki and G. S. Stavrakakis, "Optimal robot speed trajectory by minimization of the actuator motor electromechanical losses," *J. Intell. Robot Syst.*, vol. 33, pp. 187–207, 2002.

[18] S. Liu and D. Sun, "Optimal motion planning of a mobile robot with minimum energy consumption," in *Proc. of the IEEE/ASME Int. Conf. on Advanced Intell. Mechatron. (AIM)*, Budapest, Hungary, Jul. 2011, pp. 43–48.

[19] A. Trzynadlowski, "Energy optimization of a certain class of incremental motion dc drives," *IEEE Trans. Ind. Electr.*, vol. 35, no. 1, pp. 60–66, Feb. 1988.

[20] C. H. Kim and B. K. Kim, "Minimum-energy translational trajectory generation for differential-driven wheeled mobile robots," *J. Intell. Robot Syst.*, vol. 49, no. 4, pp. 367–383, 2007.

[21] H. Kim and B. K. Kim, "Online minimum-energy trajectory planning and control on a straight-line path for three-wheeled omnidirectional mobile robots," *IEEE Trans. Ind. Electr.*, vol. 61, no. 9, pp. 4771–4779, Sep. 2014.

[22] M.-S. Huang, Y.-L. Hsu, and R.-F. Fung, "Minimum-energy point-to-point trajectory planning for a motor-toggle servomechanism," *IEEE/ASME Trans. Mechatron.*, vol. 17, no. 2, pp. 337–343, 2012.

[23] J. T. Betts, *Practical Methods for Optimal Control using Nonlinear Programming*. PA: SIAM, 2001.

[24] R. F. Hartl, S. P. Sethi, and R. G. Vickson, "A survey of the maximum principles for optimal control problems with state constraints," *SIAM Review*, vol. 37, pp. 181–218, Jun. 1995.

[25] Y. Wang, Y. Zhao, and S. A. Bortoff, "A numerical algorithm to solve a class of multi-point boundary problems," in *Proc. of the 33th Chinese Control Conf.*, Xi'an, China, Jul. 2013, pp. 2262–2268.

[26] Y. Zhao and P. Tsiotras, "Density functions for mesh refinement in numerical optimal control," *J. Guid. Control Dynam.*, vol. 34, no. 1, pp. 271–277, 2011.

Yebin Wang (S'06-M'10) received the B.Eng. degree in Mechatronics Engineering from Zhejiang University, China, in 1997, M.Eng. degree in Control Theory & Control Engineering from Tsinghua University, China, in 2001, and Ph.D. in Electrical Engineering from the University of Alberta, Canada, in 2008. Dr. Wang has been with Mitsubishi Electric Research Laboratories in Cambridge, MA, USA, since 2009, and now is a Principal Member Research Staff. From 2001 to 2003 he was a Software Engineer, Project Manager, and R&D Manager in industries, Beijing, China. His research interests include nonlinear control and estimation, optimal control, adaptive systems and their applications including mechatronic systems.
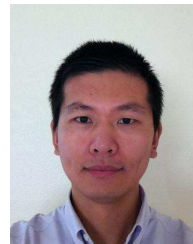
Yiming Zhao (M'12) received his Ph.D. in Aerospace Engineering (2012) and M.S. in Mathematics (2011) from Georgia Institute of Technology. He also received B.S. and M.S. degrees in Aerospace Engineering from Beijing University of Aeronautics and Astronautics, China. He has been with Halliburton, Houston, TX since 2013. Previously he was with Mitsubishi Electric Research Laboratories, Cambridge, MA. His current research interests include control and optimization of logging and drilling tools.

Scott A. Bortoff (S'83-M'92-SM'11) is currently the Group Manager of Mechatronics at Mitsubishi Electric Research Laboratories (MERL), located in Cambridge, MA, USA. Prior to joining MERL in 2009, Dr. Bortoff was the Group Manager for Controls at United Technologies Research Center, located in East Hartford, CT, USA, where he lead a group of 20 researchers and conducted industrial research and development of control for HVAC systems, fuel cell power plants and electrical power generation and distribution systems. From 2004-2009 he was an Adjunct faculty member of Rensselaer Polytechnic Institute in Hartford, CT, where he taught graduate courses in control theory. He was previously an Associate Professor at the University of Toronto, Toronto, Ontario, Canada. He earned the Ph.D. degree in Electrical Engineering from the University of Illinois at Urbana-Champaign in 1992, and undergraduate degrees in Electrical Engineering from Syracuse University, Syracuse, NY, USA. Scott's research interests include modeling and control of mechatronic systems, thermo fluid systems, and electric power systems.

Koichiro Ueda received his B.E. and M.E. degrees in applied mathematics and physics from Kyoto University, Japan, in 1996, and 1998, respectively. Since 1998, he has been with Mitsubishi Electric Corporation, Japan. His research interests include motion control and its industrial application.