Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Energy-Efficient Trajectory Planning for a Mobile Agent by Using a Two-Stage Decomposition Approach ⋆

Han Yu *, **Yebin Wang** **, **Scott A. Bortoff** ***, **Koichiro Ueda** ****

* *Department of Electrical Engineering at the University of Notre Dame, IN 46556, USA. Tel:574-339-3080; email: hyu@nd.edu.*
** *Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA 02139, USA (email: yebinwang@ieee.org)*
*** *Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA 02139, USA (email: bortoff@merl.com)*
**** *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*

Abstract: This paper presents a new approach for the energy-efficient trajectory planning of a mobile agent with obstacle avoidance. The motion of the mobile agent is subject to position constraints characterizing an obstacle (keep-out region) as well as velocity, acceleration, and control constraints. The original optimal control problem is transformed into a mathematical programming problem where the obstacle is described by a set of linear constraints and switching times, which specify the sequence of active constraints corresponding to the obstacle. A two-stages decomposition method is proposed to solve the optimal control inputs and switch times and is verified through simulations. The proposed approach can be applied to solve general obstacle avoidance trajectory planning problems.

Keywords: Optimal trajectory generation, non-convex optimization

1. INTRODUCTION

Mobile agents have been used in many applications including exploration in unknown areas, search and rescue, reconnaissance, security, military, cleaning, and personal service. For those applications, mobile agents usually carry their own power supplies such as batteries. The limited energy capacity of these carry-on power sources restrict the applications of mobile agents thus the energy efficiency of mobile agents is of importance. Energy saving of mobile agents can be achieved in several ways, for example, using energy-efficient devices (motors), energy-efficient trajectory planning etc. The energy-efficient trajectory planning is generally achieved by the path planning and the motion planning along the path. Most existing researches on the energy-efficient trajectory planning of mobile agents focus on applications to industrial robots Verscheure et al. [2008], Xu et al. [2009].

Path planning is one of the fundamental problems in control of mobile agents, and the ability to plan collision-free paths is a precondition for numerous applications of autonomous agents. Many algorithms have been proposed for solving this problem Barraquand and Latombe [1991]-Shiller [2000]. These can be roughly categorized into search-based methods, geometric approaches and probabilistic approaches. For instance, Sun and Reif [2005] studied the energy-efficient path planning problem, and the energy consumption of a mobile agent along a path was examined in terms of the friction and gravity. On the other hand, related works in optimal motion planning mostly consider time-optimality and smoothness of the trajectory. Time-optimal trajectory planning of a mobile agent was studied in Lau et al. [2009], where the generated trajectory has continuous curvature. The change of trajectory curvature was also used as the smoothness criterion in the trajectory planning Hussein and Elnagar [1997]. It was reported that the energy consumption could be minimized through optimizing the control inputs along the trajectory Guo and Tang [2008]-Howard and Kelly [2007], subject to boundary conditions of arrival time and velocity/acceleration at the start and end positions. However, these boundary conditions were designated without optimization. The energy consumption of a mobile agent with different trajectories was analyzed in Mei et al. [2004], with a proposal of an energy-efficient motion scheme. A follow-on study on the power model was reported in Mei et al. [2006]. The issue of the minimum energy control problem for three-wheel mobile agents was investigated in Kim and Kim [2008], considering the translational trajectory planning only.

This paper considers the energy-efficient trajectory planning of a mobile agent with obstacle avoidance. The obstacle is given while the motion of the mobile agent is subjective to velocity, acceleration, and control constraints. Different from conventional numerical approaches, where the original optimal control problem is directly transformed into a mixed integer or nonlinear programming problems, we manage to reformulate the non-convex constraints due to the obstacle as a set of linear constraints by introducing auxiliary decision variables 'switching time instants' thus simplify the computation of the optimal trajectory. A two-stage decomposition method is proposed to solve the optimal switching instants and inputs.

⋆ This work was done while H. Yu was an intern with Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA 02139, USA.

This approach can be generalized to solve general obstacle avoidance trajectory planning problems.

The remainder of this paper is organized as follows. In Section 2, we formulate the energy-efficient trajectory planning problem. In Section 3, we outline the two-stage decomposition approach and propose the conceptual algorithm. In Section 4, we transcribe the original optimal control problem into an equivalent problem parameterized by the switching time instants and develop a method to obtain the derivative value of the cost function with respect to the switching time instants. Examples are provided in Section V to illustrate the effectiveness of the method. Section 6 concludes the paper.

2. PROBLEM FORMULATIONS

We consider a mobile agent moving in a 2D plane and its dynamics is described by the following forth-order linear time-invariant (LTI) control system

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = d_x x_2 + b_x u_x \\ \dot{y}_1 = y_2 \\ \dot{y}_2 = d_y y_2 + b_y u_y \end{cases} \tag{1}$$

where x_1 denotes the position and x_2 denotes the velocity of the mobile agent in the x-axis, while y_1 denotes the position and y_2 denotes the velocity of the mobile agent in the y-axis. u_x, u_y are control inputs, d_x, b_x, d_y, b_y are constant. Denoting $X = [x_1, x_2, y_1, y_2]^T$ as the state of the mobile agent and $U = [u_x, u_y]^T$ as the control input, then the dynamics given in (1) can be written as follows

$$\dot{X} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & d_x & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & d_y \end{pmatrix} X + \begin{pmatrix} 0 & 0 \\ b_x & 0 \\ 0 & 0 \\ 0 & b_y \end{pmatrix} U \quad = AX + BU.$$

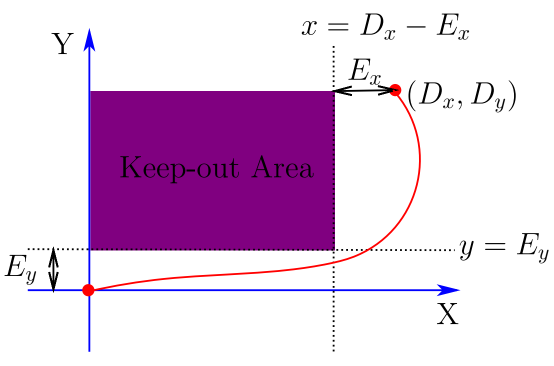

Fig. 1. Keep-Out Region of the Mobile Agent

The original trajectory planning problem is stated as follows:

Problem 1. Given the system (1) and the final arrival time t_f, design the trajectory X from Position A ($X_0 = [0,0,0,0]^T$) to Position B ($X_f = [D_x, 0, D_y, 0]^T$) with minimal cost while avoiding the keep-out area shown in Fig.1, where the cost function is given by

$$J = \int_0^{t_f} \left(R_x u_x^2 + R_y u_y^2 + K_x x_2 u_x + K_y y_2 u_y \right) dt, \tag{2}$$

with positive constants R_x, R_y, K_x, K_y, velocity and acceleration constraints are given by

$$\begin{aligned} 0 \le x_2 \le v_{max}^x, \quad 0 \le y_2 \le v_{max}^y, \\ |\dot{x}_2| \le a_{max}^x, \quad |\dot{y}_2| \le a_{max}^y, \end{aligned} \tag{3}$$

and control input constraints are given by

$$|u_x| \le u_{max}, \quad |u_y| \le u_{max}. \tag{4}$$

The cost function in Problem 1 represents the energy consumption of a mobile agent as a combination of copper and mechanical losses, which is generally a quadratic function of state and control. The method proposed in this paper however can deal with general cost functions. Also, the system considered in Problem 1 is not necessarily in the form of (1). For the rest of this paper, we will denote the admissible region for the mobile agent which can be characterized by (E_x, E_y, D_x, D_y) as shown in Fig.1 by Ω. Note that the admissible region Ω is non-convex.

2.1 *Binary Constraints for Collision Avoidance*

The obstacle avoidance imposes constraint on the position or path of the mobile agent. The resultant obstacle avoidance constraint can be formulated as:

$$x_1 \ge D_x - E_x \ \ \textbf{\textit{or}} \ \ y_1 \le E_y, \tag{5}$$

As it is well-understood, the closed-form solution of Problem 1 is difficult to establish due to variant constraints Gong et al. [2006]. We focus on the numerical computation approach, i.e., direct transcript Gong et al. [2006], Verscheure et al. [2009]. We divide the entire time t_f into a certain number of time steps, and at every time step k the position (x_1^k, y_1^k) of the mobile agent must lie in the area outside of the obstacle. This obstacle avoidance constraint (5) is rewritten as:

$$x_1^k \ge D_x - E_x \ \ \textbf{\textit{or}} \ \ y_1^k \le E_y, \tag{6}$$

where $(x_1^k,\ y_1^k)$ denotes the position of the mobile agent at the k−th time step. A way to transform the ***or***-constraint into a more useful ***and***-constraint is to introduce binary slack variables Taha [1987]. Let μ_i^k for $i = 1, 2$ be a binary variable (0 or 1) at the k−th time step and let M be an arbitrary large positive number. The constraint (6) may then be replaced by the following mixed-integer/linear constraints:

$$\begin{aligned} -x_1^k \le -(D_x - E_x) + M\mu_1^k, \\ \textbf{\textit{and}} \ \ y_1^k \le E_y + M\mu_2^k, \\ \textbf{\textit{and}} \ \ \mu_1^k + \mu_2^k \le 1, \ \ \mu_1^k, \mu_2^k \in \{0, 1\}, \end{aligned} \tag{7}$$

The last ***and***-constraint ensures that at least one of the original ***or***-constraints is satisfied. After transforming the ***or***-constraints into mixed-integer/linear constraints, Problem 1 with collision avoidance constraints (7) can be discretized as a large non-convex mixed integer/nonlinear programming (MINLP) problem. The resulting optimization problem can be readily solved by commercial programming solvers.

2.2 *Smooth Constraints for Collision Avoidance*

There are other ways to formulate constraints for obstacle avoidance, for example, the constraint (6) is equivalent to the following smooth nonlinear constraints

$$\begin{aligned} \mu_1^k(x_1^k - D_x + E_x) + \mu_2^k(y_1^k - E_y) \le 0, \\ (\mu_1^k)^2 + (\mu_2^k)^2 - 1 = 0, \\ -\mu_1^k + \mu_2^k - 1 = 0. \end{aligned} \tag{8}$$

The last two equations in (8) ensures that $(\mu_1^k,\ \mu_2^k)$ can only take two solutions: (-1, 0) and (0,1), as shown in Fig.2. One can observe that the mixed-integer/linear constraints in (7) is transformed into nonlinear constraints (8) due to the binary characterization of the parameters (μ_1^k, μ_2^k). This kind of parameterizations is not unique, for example, another alternative parameterization give the following set of smooth nonlinear constraints

$$\begin{aligned} \mu_1^k(D_x - E_x - x_1^k) + \mu_2^k(y_1^k - E_y) \le 0, \\ \mu_1^k \mu_2^k = 0, \\ \mu_1^k + \mu_2^k - 1 = 0. \end{aligned} \tag{9}$$

The last two equations in (9) ensures that (μ_1^k, μ_2^k) can only take two solutions: (1, 0) and (0,1), as shown in Fig.3. One can readily verify that constraints (9) are equivalent to (6) in the sense that both constraints lead to the same admissible domain Ω of position variables.

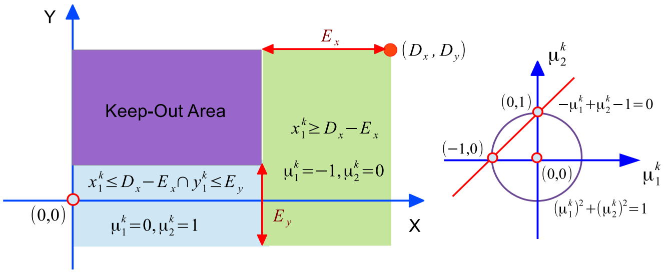

Fig. 2. Alternative Parameterization of the Constraint (8)

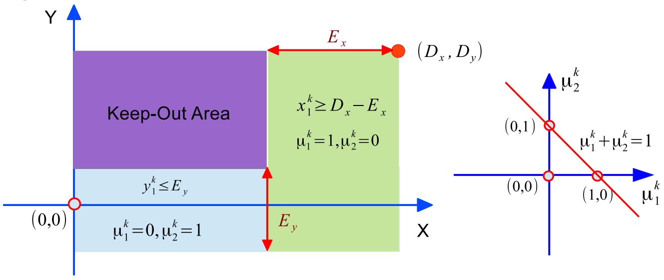

Fig. 3. Alternative Parameterization of the Constraint (9)

Note that Problem 1 with the smooth nonlinear constraints for collision avoidance (8) or (9) can be transcribed to a large non-convex nonlinear programming (NLP) problem. The resultant optimization problem can be solved by a nonlinear programming solver, i.e., **fmincon** in MATLAB.

3. TWO-STAGE DECOMPOSITION APPROACH

3.1 Characterizing Collision Avoidance by Switching Time Instants of the Active Constraints

As shown in Section 2, direct transcription of Problem 1 often leads to MINLP or NLP problems which are computational intensive and suffers the feasibility issue. We try to simplify the computation by taking further look into constraints (3) and reformulating Problem 1. Specifically, the positive velocity constraint in x-axis implies that the optimal position trajectory of the mobile agent can cross the line $x = D_x - E_x$ once. That is: given any optimal trajectory, there exists a time instant t_s such that, the first segment of the optimal position trajectory lies in the region $x_1(t) \le D_x - E_x$ for $0 \le t \le t_s$, and the second segment of the optimal position trajectory lies in the region $x_1(t) \ge D_x - E_x$ for $t_s \le t \le t_f$. More precisely, the optimal trajectory switches once from the admissible region given by

$$\textbf{Region 1:} \begin{cases} 0 \le x_2 \le \upsilon_{max}^x, \quad 0 \le y_2 \le \upsilon_{max}^y, \\ |\dot{x}_2| \le a_{max}^x, \quad |\dot{y}_2| \le a_{max}^y, \\ |u_x| \le u_{max}, \quad |u_y| \le u_{max}, \\ \dot{X} = AX + BU, \quad X(0) = X_0, \\ y_1 \le E_y, \quad 0 \le t < t_s, \end{cases} \tag{10}$$

to the admissible region given by

$$\textbf{Region 2:} \begin{cases} 0 \le x_2 \le \upsilon_{max}^x, \quad 0 \le y_2 \le \upsilon_{max}^y, \\ |\dot{x}_2| \le a_{max}^x, \quad |\dot{y}_2| \le a_{max}^y, \\ |u_x| \le u_{max}, \quad |u_y| \le u_{max}, \\ \dot{X} = AX + BU, \quad X(t_s^+) = X(t_s^-), \quad X(t_f) = X_f, \\ x_1 \ge D_x - E_x, \quad t_s \le t \le t_f, \end{cases} \tag{11}$$

where t_s denotes the time instant at which the optimal trajectory enters Region 2 from Region 1. Problem 1 is loosely reformulated as follows

Problem 2. Given the system (1) and the final time t_f, find the optimal control input U^* and the optimal switching time t_s^* such that the corresponding continuous state trajectory X departing from a given initial state $X(t_0) = X_0$ meets all the constraints in Region 1 and Region 2 respectively and arrives at X_f at time t_f, while the cost function J given by (2) is minimized.

Remark 3. After introducing the switching time instant t_s, the non-convex admissible region Ω for the mobile agent is now split into two convex sub-regions $y_1 \le E_y$ (for $0 \le t < t_s$) and $x_1 \ge D_x - E_x$ (for $t_s \le t < t_f$), Problem 1 is reduced to solve the optimal control problem with the admissible region described by (10) and (11), and the switching time instant t_s. Given a fixed t_s, constraints in Problem (2) are linear.

Remark 4. Although it is still challenging to apply the indirect method to Problem 2, introducing the switch time instant t_s does simplify the derivation of necessary conditions from the minimum principle Bryson and Ho [1975]. Necessary conditions could be potentially useful to get insight on the properties of the optimal solutions as Wang et al. [2012, 2013].

3.2 Two-Stages Decomposition

We can decompose *Problem 2* into two stages. **Stage 1** is to solve a conventional optimal control problem for U which minimizes the cost function J under a given switching time t_s. We denote the corresponding optimal cost function as $J(t_s)$. **Stage 2** is trying to minimize the cost function $J(t_s)$ with respect to t_s (i.e., $\min_{t_s} J(t_s)$, subject to $0 < t_s < t_f$). The conceptual algorithm is stated as follows:

(1) Set the iteration index $j = 0$, choose an initial t_s^j.
(2) By solving an optimal control problem (i.e., **Stage 1**), calculate $J(t_s^j)$.
(3) Calculate $\frac{\partial J}{\partial t_s}|_{t_s^j}$.
(4) Use some feasible direction method to update t_s^j to be $t_s^{j+1} = t_s^j + \alpha^j dt_s^j$ (here dt_s^j is formed by using the gradient information of J with respect to t_s; the step size α^j can be chosen using some step size rule). Set the iteration index $j = j+1$.
(5) Repeat step 2)-4) until a prescribed termination condition is satisfied.

Remark 5. Decomposition of Problem 2 into two stages is motivated by the fact that given a fixed t_s, the obstacle constraints are convex thus the corresponding optimal control problem can be solved more efficiently and reliably. The decomposition is particularly effective for the case when the cost function is convex. The **Stage 1** of Problem 2 can be accomplished using numerical computation techniques. Differently, **Stage 2** requires at least the knowledge of gradient of the cost function J with respect to the switch time t_s, whose analytical expression, except for very few classes of problems, are almost impossible to obtain. However, we can numerically compute the value of the derivative $\frac{\partial J}{\partial t_s}$ from integrating sensitivity equations derived in the next section.

4. EQUIVALENT PROBLEM FORMULATION BASED ON PARAMETERIZATION OF THE SWITCHING INSTANTS

We now describe the transcription of Problem 2 into an equivalent problem parameterized by unknown switching instants. For Problem 2, only one switch time t_s is required. Thus we introduce a variable z which corresponds to the switching time t_s. Let z satisfy

$$\begin{cases} \dfrac{dz}{dt} = 0 \\ z(0) = t_s. \end{cases} \tag{12}$$

Next, introduce a new independent time variable τ, a piecewise linear relationship between t and τ is established as

$$t = \begin{cases} t_0 + (z - t_0)\tau, & 0 \le \tau \le 1, \\ z + (t_f - z)(\tau - 1), & 1 \le \tau \le 2. \end{cases} \tag{13}$$

The expression of Region 1 in τ time scale is summarized as follows: for $\tau \in [0,1]$, the dynamics of the mobile agent is

$$\begin{cases} \dfrac{dx_1(\tau)}{d\tau} = (z - t_0)x_2(\tau), \quad x_1(0) = 0, \\ \dfrac{dx_2(\tau)}{d\tau} = (z - t_0)[d_x x_2(\tau) + b_x u_x(\tau)], \quad x_2(0) = 0, \\ \dfrac{dy_1(\tau)}{d\tau} = (z - t_0)y_2(\tau), \quad y_1(0) = 0, \\ \dfrac{dy_2(\tau)}{d\tau} = (z - t_0)[d_y y_2(\tau) + b_y u_y(\tau)], \quad y_2(0) = 0, \\ \dfrac{dz(\tau)}{d\tau} = 0, \quad z(0) = t_s, \end{cases} \tag{14}$$

while constraints on the velocity, acceleration and control inputs are given by

$$\begin{cases} 0 \le x_2(\tau) \le \upsilon^x_{max}, \;\; 0 \le y_2(\tau) \le \upsilon^y_{max}, \\ |\dot{x}_2(\tau)| \le a^x_{max}, \;\; |\dot{y}_2(\tau)| \le a^y_{max}, \\ |u_x(\tau)| \le u_{max}, \quad |u_y(\tau)| \le u_{max}, \\ y_1(\tau) \le E_y. \end{cases} \tag{15}$$

Region 2 in τ time scale is summarized as follows: for $\tau \in [1,2]$, the dynamics of the mobile agent is given by

$$\begin{cases} \dfrac{dx_1(\tau)}{d\tau} = (t_f - z)x_2(\tau), \quad x_1(2) = D_x, \\ \dfrac{dx_2(\tau)}{d\tau} = (t_f - z)[d_x x_2(\tau) + b_x u_x(\tau)], \quad x_2(2) = 0, \\ \dfrac{dy_1(\tau)}{d\tau} = (t_f - z)y_2(\tau), \quad y_1(2) = D_y, \\ \dfrac{dy_2(\tau)}{d\tau} = (t_f - z)[d_y y_2(\tau) + b_y u_y(\tau)], \quad y_2(2) = 0, \\ \dfrac{dz(\tau)}{d\tau} = 0, \quad z(1) = t_s, \end{cases} \tag{16}$$

while constraints on the velocity, acceleration and the control inputs are given by

$$\begin{cases} 0 \le x_2(\tau) \le \upsilon^x_{max}, \;\; 0 \le y_2(\tau) \le \upsilon^y_{max}, \\ |\dot{x}_2(\tau)| \le a^x_{max}, \;\; |\dot{y}_2(\tau)| \le a^y_{max}, \\ |u_x(\tau)| \le u_{max}, \quad |u_y(\tau)| \le u_{max} \\ x_1(\tau) \ge D_x - E_x. \end{cases} \tag{17}$$

In the τ time scale, the cost function J shown in (2) now becomes

$$\begin{aligned} \widetilde{J} &= \widetilde{J}_1 + \widetilde{J}_2 \\ &= \int_0^1 (z - t_0)(R_x u_x^2 + R_y u_y^2 + K_x x_2 u_x + K_y y_2 u_y) d\tau \\ &+ \int_1^2 (t_f - z)(R_x u_x^2 + R_y u_y^2 + K_x x_2 u_x + K_y y_2 u_y) d\tau. \end{aligned} \tag{18}$$

Problem 2 is equivalent to the following problem.

Problem 6. Find the optimal switching time $z(\tau)$ and the optimal control input $U(\tau) = [u_x(\tau), u_y(\tau)]^T$ for $\tau \in [0,2]$ such that the corresponding continuous state trajectory X departing from a given initial state $X(t_0) = X_0$ meets all constraints in Regions 1 and 2 given by (15) and (17) respectively and arrives at X_f at time t_f, while the cost function J given by (18) is minimized.

It should be noted that Problems 2 and 6 are equivalent in the sense that an optimal solution for Problem 6 is an optimal solution for Problem 2 by a proper change of variable as shown in (13) and *vice versa*.

Based on Problem 6, we now develop a method for computing the numerical value of $\frac{\partial \widetilde{J}}{\partial z}$. Let $L(\tau) = R_x u_x^2(\tau) + R_y u_y^2(\tau) + K_x x_2(\tau)u_x(\tau) + K_y y_2(\tau)u_y(\tau)$, then we have

$$\begin{aligned} \frac{\partial \widetilde{J}}{\partial z} &= \frac{\partial \widetilde{J}_1}{\partial z} + \frac{\partial \widetilde{J}_2}{\partial z} \\ &= \int_0^1 \left\{ L + (z - t_0)\left[\left(\frac{\partial L}{\partial X}\right)^T \frac{\partial X}{\partial z} + \left(\frac{\partial L}{\partial U}\right)^T \frac{\partial U}{\partial z}\right]\right\} d\tau \\ &+ \int_1^2 \left\{ -L + (t_f - z)\left[\left(\frac{\partial L}{\partial X}\right)^T \frac{\partial X}{\partial z} + \left(\frac{\partial L}{\partial U}\right)^T \frac{\partial U}{\partial z}\right]\right\} d\tau, \end{aligned}$$

where

$$\begin{aligned} \left(\frac{\partial L}{\partial X}\right)^T \frac{\partial X}{\partial z} &= K_x u_x \frac{\partial x_2}{\partial z} + K_y u_y \frac{\partial y_2}{\partial z}, \\ \left(\frac{\partial L}{\partial U}\right)^T \frac{\partial U}{\partial z} &= (2R_x u_x + K_x x_2)\frac{\partial u_x}{\partial z} + (2R_y u_y + K_y y_2)\frac{\partial u_y}{\partial z}. \end{aligned}$$

We need to know $\frac{\partial x_2}{\partial z}$, $\frac{\partial y_2}{\partial z}$, $\frac{\partial u_x}{\partial z}$ and $\frac{\partial u_y}{\partial z}$ in order to get $\frac{\partial \widetilde{J}}{\partial z}$.

For $\tau \in [0,1]$, we have

$$\begin{aligned} \frac{\partial}{\partial \tau}\Big(\frac{\partial x_2}{\partial z}\Big) &= \frac{\partial}{\partial z}\Big(\frac{\partial x_2}{\partial \tau}\Big) \\ &= \frac{\partial}{\partial z}\Big((z - t_0)(d_x x_2 + b_x u_x)\Big) \\ &= d_x x_2 + b_x u_x + (z - t_0)d_x \frac{\partial x_2}{\partial z} + (z - t_0)b_x \frac{\partial u_x}{\partial z}, \\ \frac{\partial}{\partial \tau}\Big(\frac{\partial y_2}{\partial z}\Big) &= \frac{\partial}{\partial z}\Big(\frac{\partial y_2}{\partial \tau}\Big) \\ &= \frac{\partial}{\partial z}\Big((z - t_0)(d_y y_2 + b_y u_y)\Big) \\ &= d_y y_2 + b_y u_y + (z - t_0)d_y \frac{\partial y_2}{\partial z} + (z - t_0)b_y \frac{\partial u_y}{\partial z}, \end{aligned} \tag{19}$$

where $\frac{\partial u_x}{\partial z}$ can be obtained via

$$\begin{cases} \dfrac{\partial \widetilde{J}_1}{\partial u_x} = \displaystyle\int_0^1 (z-t_0)(2R_x u_x + K_x x_2) d\tau = 0, \\ \dfrac{\partial}{\partial z}\dfrac{\partial \widetilde{J}_1}{\partial u_x} = 0. \end{cases} \quad (20)$$

Equation (20) yields

$$\frac{\partial u_x}{\partial z} = \frac{1}{2R_x}\Big(-\frac{2R_x u_x + K_x x_2}{z-t_0} - K_x \frac{\partial x_2}{\partial z}\Big). \quad (21)$$

Similarly we can get

$$\frac{\partial u_y}{\partial z} = \frac{1}{2R_y}\Big(-\frac{2R_y u_y + K_y y_2}{z-t_0} - K_y \frac{\partial y_2}{\partial z}\Big). \quad (22)$$

From (19)-(22), we can get the numerical value of $\frac{\partial \widetilde{J}_1}{\partial z}$. For $\tau \in [1,2]$, we have

$$\begin{aligned} \frac{\partial}{\partial \tau}\Big(\frac{\partial x_2}{\partial z}\Big) &= \frac{\partial}{\partial z}\Big(\frac{\partial x_2}{\partial \tau}\Big) \\ &= \frac{\partial}{\partial z}\Big((t_f - z)(d_x x_2 + b_x u_x)\Big) \\ &= -d_x x_2 - b_x u_x + (t_f - z) d_x \frac{\partial x_2}{\partial z} + (t_f - z) b_x \frac{\partial u_x}{\partial z}, \\ \frac{\partial}{\partial \tau}\Big(\frac{\partial y_2}{\partial z}\Big) &= \frac{\partial}{\partial z}\Big(\frac{\partial y_2}{\partial \tau}\Big) \\ &= \frac{\partial}{\partial z}\Big((t_f - z)(d_y y_2 + b_y u_y)\Big) \\ &= -d_y y_2 - b_y u_y + (t_f - z) d_y \frac{\partial y_2}{\partial z} + (t_f - z) b_y \frac{\partial u_y}{\partial z}, \end{aligned} \quad (23)$$

where $\frac{\partial u_x}{\partial z}$ can be obtained via

$$\begin{cases} \dfrac{\partial \widetilde{J}_2}{\partial u_x} = \displaystyle\int_1^2 (t_f - z)(2R_x u_x + K_x x_2) d\tau = 0, \\ \dfrac{\partial}{\partial z}\dfrac{\partial \widetilde{J}_1}{\partial u_x} = 0. \end{cases} \quad (24)$$

Equation (24) yields

$$\frac{\partial u_x}{\partial z} = \frac{1}{2R_x}\Big(\frac{2R_x u_x + K_x x_2}{t_f - z} - K_x \frac{\partial x_2}{\partial z}\Big). \quad (25)$$

Similarly we can get

$$\frac{\partial u_y}{\partial z} = \frac{1}{2R_y}\Big(\frac{2R_y u_y + K_y y_2}{t_f - z} - K_y \frac{\partial y_2}{\partial z}\Big). \quad (26)$$

Given (23)-(26), we can compute the value of $\frac{\partial \widetilde{J}_2}{\partial z}$, consequentially, $\frac{\partial \widetilde{J}}{\partial z}$ through the calculations of $\frac{\partial \widetilde{J}_1}{\partial z}$ and $\frac{\partial \widetilde{J}_2}{\partial z}$.

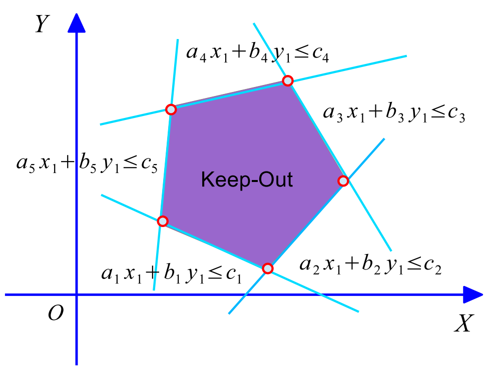

Fig. 4. Polyhedron Obstacle

Remark 7. It can be seen that there is no difficulty in applying the proposed method to energy efficient trajectory planning problems where the obstacle could be described by a polyhedron in the 2D plan (see Fig.4). In this case, we will have more than one switchings, i.e., the admissible regions for the mobile agent can be divided as

$$\begin{cases} a_1x_1 + b_1y_1 \le c_1 \ \ (\text{for } t \in [0, t_{s_1})) \\ a_2x_1 + b_2y_1 \le c_2 \ \ (\text{for } t \in [t_{s_2}, t_{s_3})) \\ a_3x_1 + b_3y_1 \le c_3 \ \ (\text{for } t \in [t_{s_3}, t_{s_4})) \\ \qquad \vdots \\ a_Kx_1 + b_Ky_1 \le c_K \ \ (\text{for } t \in [t_{s_K}, t_f)), \end{cases} \quad (27)$$

where $a_i, b_i, c_i \in \mathbb{R}$, for $i = 1, 2, \ldots, K$.

For this more general obstacle avoidance trajectory planning problem, first of all, we can transcribe the problem into an equivalent problem in $\tau \in [0, K+1]$ where K denotes the total number of switches. It is then straightforward to use the two-stages decomposition method discussed in Section 3, where the conceptual algorithm will be rectified as

(1) Set the iteration index $j = 0$, choose an initial $t_s^j = [t_{s_1}^j, t_{s_2}^j, \ldots, t_{s_K}^j]$.
(2) By solving an optimal control problem (i.e., **Stage 1**), calculate $J(t_s^j)$.
(3) Calculate $\frac{\partial J}{\partial t_s}|_{t_s^j} = \Big[\frac{\partial J}{\partial t_{s_1}^j}, \frac{\partial J}{\partial t_{s_2}^j}, \ldots, \frac{\partial J}{\partial t_{s_K}^j}\Big]$.
(4) Use some feasible direction method to update t_s^j to be $t_s^{j+1} = t_s^j + \alpha^j dt_s^j$ (here dt_s^j is formed by using the gradient information of J with respect to t_s; the step size α^j can be chosen using some step size rule). Set the iteration index $j = j+1$.
(5) Repeat step 2)-4) until a prescribed termination condition is satisfied.

As it is clear from Remark 7, the generalization of the proposed method to more general obstacle cases relies on the partition of the admissible domain into a union of convex domains, and the determination of the order of convex domains through which a path passes. How a path passes through these convex domains is required as a priori to apply the proposed approach.

5. EXAMPLES

We consider a mobile agent with dynamics given by (1), where $d_x = 6.33$, $b_x = 2834.3$, $d_y = 6.42$, $b_y = 1093.7$, $\upsilon_{max}^x = 2.499$, $\upsilon_{max}^y = 2.499$, $a_{max}^x = 103.5$, $a_{max}^y = 79.6$, $u_{max} = 7.01$. The keep-out region is characterized by $D_x = 2.0$, $D_y = 3.0$, $E_x = 0.05$, $E_y = 0.05$. The final arrival time is given by $t_f = 2.855$s. Using the proposed two-stage decomposition approach discussed in Section 4 and using MATLAB function **fmincon** to solve the nonlinear programming(NLP) problem in stage 1 (the optimal control problem in stage 1 can be transformed into a NLP by applying collocation method), we have the simulation results shown in Fig.5. The initial guess for the switching time is $t_s^0 = 0.8064$s, and the resultant cost $\widetilde{J}(t_s^0) = 236.92$J. The optimal switching time obtained by applying the proposed algorithm is $t_s^* = 0.8373$s with $\widetilde{J}(t_s^*) = 235.97$J. Compared to the NLP result from the smooth collision avoidance constraints in Section 2, the optimization problem corresponding to Problem 6 takes less time to compute the solution.

Now change the keep-out region to be characterized by $D_x = 0.5$, $D_y = 0.5$, $E_x = 0.1$, $E_y = 0.1$. Set the final arrival time to be $t_f = 0.455$s, we have the simulation results shown in Fig.6. The initial guess for the switching time is $t_s^0 = 0.1742$s, and the resultant cost $\widetilde{J}(t_s^0) = 57.47$J. The optimal switching time obtained by applying the proposed algorithm is $t_s^* = 0.1694$s with $\widetilde{J}(t_s^*) = 45.76$J.

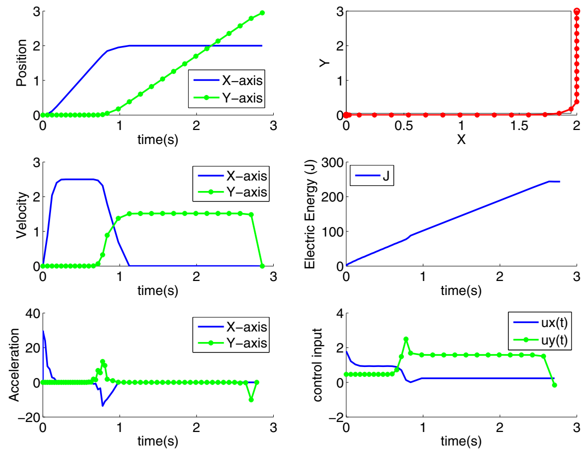

Fig. 5. $D_x = 2.0$, $D_y = 3.0$, $E_x = 0.05$, $E_y = 0.05$

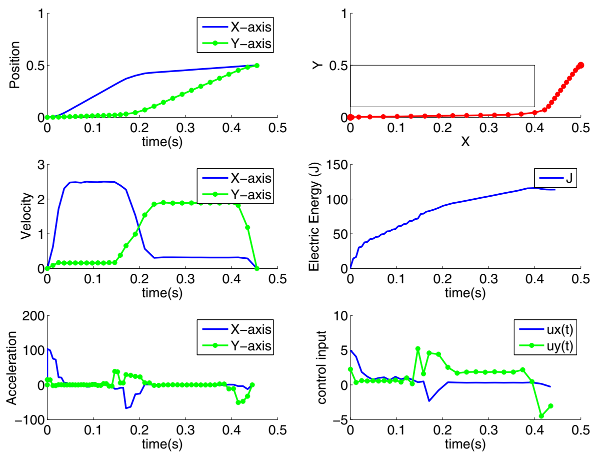

Fig. 6. $D_x = 0.5$, $D_y = 0.5$, $E_x = 0.1$, $E_y = 0.1$

6. CONCLUSION

In this paper, we present a new approach for energy-efficient trajectory planning of mobile agents with obstacle avoidance. The original optimal control problem is transformed into a mathematical programming problem where the keep-out region is described by a set of linear constraints and switching time instants. A two-stages decomposition method is applied to solve this problem and is verified through simulations. This approach can be applied to solve general obstacle avoidance path planning problems.

REFERENCES

J. Barraquand and J.-C. Latombe. Nonholonomic multibody mobile robots: Controllability and motion planning in the presence of obstacles. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 2328–2335, 1991.

Jr. Arthur E. Bryson and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation and Control*. Taylor & Francis, 1975.

Q. Gong, W. Kang, and I. M. Ross. A pseudospectral method for the optimal control of constrained feedback linearizable systems. *IEEE Trans. Automat. Contr.*, 51(7):1115–1129, Jul. 2006.

Y. Guo and T. Tang. Optimal trajectory generation for nonholonomic robots in dynamic environments. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 2552–2557, 2008.

T. Howard and A. Kelly. Optimal rough terrain trajectory generation for wheeled mobile robots. *The International Journal of Robotics Research*, 26:141–166, 2007.

A.M. Hussein and A. Elnagar. On smooth and safe trajectory planning in 2d environments. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 3118–3123, 1997.

H. Kim and B.-K. Kim. Minimum-energy translational trajectory planning for battery-powered three-wheeled omni-directional mobile robots. In *Proc. of IEEE Int. Conf. on Control, Automation, Robotics and Vision*, pages 1730–1735, 2008.

B. Lau, C. Sprunk, and W. Burgard. Kinodynamic motion planning for mobile robots using splines. In *Proc. of IEEE Int. Conf. on Intelligent Robots and Systems*, pages 2427–2433, 2009.

Y. Mei, Y. Lu, Y. Hu, and C. Lee. Energy-efficient motion planning for mobile robots. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 4344–4349, 2004.

Y. Mei, Y. Lu, Y. Hu, and C. Lee. Deployment of mobile robots with energy and timing constraints. *IEEE Transactions on Robotics*, 22(3):507–522, Jun. 2006.

Z. Shiller. Obstacle traversal for space exploration. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 989–994, 2000.

Z. Sun and J. Reif. On finding energy-minimizing paths on terrains. *IEEE Transactions on Robotics*, 21(1):102–114, Feb. 2005.

Hamdy A. Taha. *Operations Research, An Introduction*. Macmillan Publishing Company, New York, 1987.

D. Verscheure, B. Demeulenaere, J. Swevers, J. De Schutter, and M. Diehl. Time-energy optimal path tracking for robots: a numerically efficient optimization approach. In *Proc. of 10th IEEE Int. Workshop on Advanced Motion Control*, pages 727–732, 2008.

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. Contr.*, 54(10): 2318–2327, Oct. 2009.

Y. Wang, K. Ueda, and S. A. Bortoff. On the optimal trajectory generation for servomotors: a Hamiltonian approach. In *Proc. 51th CDC*, pages 7620–7625, Maui, HI, Dec. 2012.

Y. Wang, K. Ueda, and S. A. Bortoff. A Hamiltonian approach to compute an energy efficient trajectory for a servomotor system. *Automatica*, 49(12):3550–3561, Dec. 2013.

H. Xu, J. Zhuang, S. Wang, and Z. Zhu. Global time-energy optimal planning of robot trajectories. In *Proc. of Int. Conf. on Mechatronics and Automation*, pages 4034–4039, 2009.