The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Smooth Proximity Computation for Collision-Free Optimal Control of Multiple Robotic Manipulators

J. Cascio, M. Karpenko, Q. Gong, P. Sekhavat, and I. M. Ross

Abstract— **This paper presents a novel approach for trajectory planning of multiple robot manipulators operating amongst obstacles. Karush-Kuhn-Tucker (KKT) conditions are exploited to compute the proximity between line-swept sphere (LSS) bounding volumes used to model potentially colliding objects. The KKT multipliers and the parameters giving the minimum distance between LSS volumes are augmented into the manipulator trajectory planning problem as dummy control variables. These extra variables allow the planning problem to be cast as a standard nonlinear optimal control problem with smooth path constraints, which is then solved using the pseudospectral method. The utility of the approach is demonstrated by a trajectory planning example involving stationary workspace obstacles and for a centralized multi-robot system in which each robot acts as a dynamic obstacle that the other should avoid. The optimal control formulation incorporates practical constraints on the manipulator joint angles, velocities and accelerations as well as limits on the control torque. The computed collision-free optimal trajectories are executed on a pair of experimental robots to verify the feasibility of the numerical results.**

I. Introduction

Computing optimal and collision-free joint trajectories for robotic manipulators operating in the presence of static and dynamic obstacles remains an open and widely studied problem in the robotics literature. A key element in developing an optimization algorithm for generating collision-free trajectories is a means for evaluating the proximity of two potentially colliding objects. The Euclidean *separation distance*, or shortest line segment joining two objects, is a natural measure of this proximity [1]. Some of the most general and versatile algorithms for computing separation distance are based on the concept of bounding volumes [2], in which potentially colliding objects are modeled using simple geometric primitives. The notion of bounding volumes, especially spherical bounding volumes, has been extensively employed in the development of path planners for robotic manipulators (see for example [3]–[5]). Sphere-swept volumes, such as the line-swept sphere (LSS), give another approach for modeling the manipulator and obstacles with increased accuracy [6].

In addition to the problem of avoiding collisions between objects, robot trajectory planning also involves solving joint motion and control profiles that maximize performance, subject to kinematic, dynamic and control constraints. This adds complexity to the planning task because it is necessary not only to find feasible collision-free paths for the robot to follow, but also to select the one that gives the optimal performance. Many authors have favored direct approaches to solving this optimal control problem. Direct methods convert the original problem into a nonlinear programming (NLP) problem that can be solved numerically.

A common approach for reformulating the original trajectory planning problem into a NLP is to discretize the joint trajectories by splining together low-order polynomials. Cubic polynomials [7], [8], clamped-cubic splines [4], and B-splines [9] have all been used for parameterizing the joint variables. Pseudospectral (PS) optimal control methods [10]–[13] have also been recently applied to discretize the trajectory planning problem for a single robot [14]. PS methods can be advantageous as they exploit the properties of orthogonal polynomials evaluated on a set of non-uniformly distributed quadrature nodes to improve the rate of convergence towards the actual continuous time optimal solution.

The objective of this paper is to present a novel approach for manipulator trajectory planning in the presence of static and dynamic obstacles and demonstrate its utility on an experimental multi-arm robotic system. In particular, a scheme is developed that integrates separation distance computations into the PS optimal control framework, in a smooth fashion. This is in contrast to the plethora of other approaches for computation of separation distance between LSS objects that rely on the evaluation of nested logical structures for implementation [1], [15], [16]. In the optimal control framework, use of these latter methods leads to a problem formulation that is inherently more difficult to solve. The difficulty arises because the Jacobian information required by the NLP solver to find the search direction can change erratically due to the non-smooth nature of the nested logical operands.

To formulate the trajectory planning problem as a smooth optimal control problem, we treat the separation distance computations as minimization sub-problems with inequality constraints. The Karush-Kuhn-Tucker (KKT) conditions [17] are exploited to write the minimum distance, for each potential collision, in terms of a set of KKT multipliers. These KKT multipliers, along with the parameters defining the minimum distance between objects, are introduced as augmented control variables. The number of decision variables increases linearly as part of this process. However, non-smoothness of the separation distance constraints is eliminated entirely by utilizing the additional control variables to lift the di-

This work was supported, in part, by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

J. Cascio, M. Karpenko, P. Sekhavat, and I. M. Ross are with the Department of Mechanical and Astronautical Engineering, Naval Postgraduate School, Monterey, CA, 93943, USA

Q. Gong is with the Department of Applied Mathematics & Statistics, University of California, Santa Cruz, CA, 95064, USA

mension of the optimization problem. The efficacy of the proposed approach is demonstrated by trajectory planning experiments involving stationary workspace obstacles and for a centralized multi-robot system in which each robot acts as a dynamic obstacle that the other should avoid. Constraints on the manipulator joint angles, velocities and accelerations as well as limits on the control torque are considered to ensure feasibility of the results.

II. Problem Statement

A. Manipulator Dynamics

The dynamics of a rigid n-link robotic manipulator are given by the Euler-Lagrange equations of motion [18]

$$\tau = M(q)\ddot{q} + C(q,\dot{q}) + G(q) \tag{1}$$

where $q \in R^n$ is the vector of joint variables. Inertia matrix $M(q)$ is an n-dimensional square, symmetric, and positive definite matrix. $C(q,\dot{q})$ is the vector of centrifugal and Coriolis terms and vector $G(q)$ captures the gravitational effects. The control torques applied by the actuators are given by τ.

The inertia matrix is calculated as

$$M(q) = \sum_{i=1}^{n} \left(J_{v_{C,i}}^T m_i J_{v_{C,i}} + J_{\omega_{C,i}}^T R_i I_i R_i^T J_{\omega_{C,i}} \right) \tag{2}$$

where m_i and I_i are the mass and inertia tensor for link i. Matrix R_i is the link rotation matrix and matrices, $J_{v_{C,i}}$ and $J_{\omega_{C,i}}$ are the link Jacobian submatrices.

The n elements, C_k, of the centrifugal/Coriolis vector are calculated from

$$C_k(q,\dot{q}) = \sum_{i=1}^{n}\sum_{j=1}^{n} \Gamma_{i,j,k}\dot{q}_j\dot{q}_i \tag{3}$$

where $\Gamma_{i,j,k}$ are the Christoffel symbols [18]

$$\Gamma_{i,j,k} = \frac{1}{2}\left(\frac{\partial M_{k,j}}{\partial q_i} + \frac{\partial M_{k,i}}{\partial q_j} - \frac{\partial M_{i,j}}{\partial q_k} \right) \tag{4}$$

The j^{th} element of the gravity vector is

$$G_j(q) = \sum_{i=1}^{n} m_i g_i^T J_{v_{C,i}}^j \tag{5}$$

where g_i denotes the gravitational vector with respect to link i.

B. Optimal Control Problem

The objective of the optimal trajectory planning problem is to synthesize joint motion and control profiles that maneuver the robot arm between two locations in the workspace while avoiding obstacles and minimizing a performance index subject to physical constraints and actuator limits. In this paper, the overall maneuver time, t_f, is minimized. Other measures of performance, such as control effort or energy consumption, can be easily accommodated simply by changing the objective function.

Selecting $\mathbf{x}^T = [q^T \,|\, \dot{q}^T]$ and $\mathbf{u} = [\ddot{q}^T]$, the optimal control problem is defined as

$$\begin{array}{ll} \text{minimize} & J[\mathbf{x},\mathbf{u},t_f] = t_f \\ \text{subject to} & \dot{\mathbf{x}}(t) = f\,(\mathbf{x},\mathbf{u},t) \\ & \mathbf{x}(t_0) = \mathbf{x}_0 \\ & \mathbf{e}\,(\mathbf{x}_f, t_f) = \mathbf{0} \\ & q^L \le q(t) \le q^U \\ & \dot{q}^L \le \dot{q}(t) \le \dot{q}^U \\ & \ddot{q}^L \le \ddot{q}(t) \le \ddot{q}^U \\ & \tau^L \le \tau(t) \le \tau^U \\ & \delta_{\min}^2 - \delta_{i,j}(t)^2 \le 0 \end{array} \tag{6}$$

In (6), $\dot{\mathbf{x}}$ denotes the set of first-order differential equations describing the dynamics of the manipulator, derived from (1). The initial conditions are given by $\mathbf{x}_0$ while $\mathbf{x}_f$ denotes the terminal state. The kinodynamic limits are enforced by the box-constraints on the joint variables and control torques. Obstacle avoidance is included by inequality $\delta_{\min}^2 - \delta_{i,j}(t)^2 \le 0$ where $\delta_{i,j}(t)$ and $\delta_{\min}$ refer to the actual separation distance and specified minimum distance threshold between any two potentially colliding objects. Integration of smooth constraints on obstacle avoidance into the optimal control problem is elaborated on in section III.

C. Pseudospectral Discretization

The theory underlying the Legendre PS optimal control has been well documented elsewhere [11]–[13]. Therefore, this section presents only a brief overview of the PS discretization in relation to optimal control of robotic manipulators.

In the Legendre PS method, the joint trajectories, $q(t)$, are approximated by an N-th order Lagrange interpolating polynomial, $q^N(t)$, evaluated at the Legnedre-Gauss-Lobatto (LGL) nodes. The distribution of the LGL nodes is non-uniform over the interval $[-1,1]$ with a high density of nodes near the end points. This characteristic of the PS discretization effectively inhibits the Runge phenomena, and improves the rate of convergence towards the actual continuous time optimal solution.

Letting parameter $\bar{q}_k^N$ be the approximation of the joint trajectory at node t_k, the time history of the approximate joint motion is given by

$$q(t) \approx q^N(t) = \sum_{k=0}^{N} \bar{q}_k^N \phi_k(t) \tag{7}$$

where $\phi_k(t)$ is the Lagrange interpolating polynomial defined as

$$\phi_k(t) = \frac{1}{N(N+1)L_N(t_k)} \frac{(t^2-1)\dot{L}_N(t)}{t-t_k} \tag{8}$$

In (8), $L_N(t)$ and $\dot{L}_N(t)$ are the N-th order Legendre polynomial and its time derivative, respectively. Since the LGL nodes, t_k, are the roots of $\dot{L}_N(t)$, it can be shown that $\phi_k(t_j) = 1$ if $k = j$ and $\phi(t_j) = 0$ if $k \neq j$. This gives the desirable property that the joint constraints translate directly to limits on the trajectory parameters that are varied by the NLP solver.

The approximated joint velocities and accelerations are computed from the derivatives of $q_i^N(t)$ at LGL node t_k. For example, the joint velocity is given by

$$\dot{q}(t) \approx q_i^N(t_k) = \sum_{j=0}^{N} \mathbf{D}_{kj} \bar{q}_i^N(t_j), \quad i = 1, 2, \ldots, n \tag{9}$$

where $\mathbf{D}$ is a fixed $N+1 \times N+1$ differentiation matrix having elements

$$\mathbf{D}_{kj} = \frac{2}{t_f - t_0} \begin{cases} \frac{L_N(\tau_k)}{L_N(\tau_j)} \frac{1}{\tau_k - \tau_j}, & \text{if } k \neq j; \\ -\frac{N(N+1)}{4}, & \text{if } k = j = 0; \\ \frac{N(N+1)}{4}, & \text{if } k = j = N; \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

Given the PS approximation of the joint trajectory and its derivatives, the path constraints on the joint torque profiles can be evaluated directly by solving (1) at each node. Moreover, since Lagrange polynomials converge rapidly to actual extremal joint trajectories and Lagrange interpolation can be applied exactly between the nodes, it is easy to verify whether the continuous joint trajectories are in fact collision-free.

III. Obstacle Avoidance

The robot links and the workspace obstacles are modeled using line-swept sphere (LSS) bounding volumes as shown in Fig. 1. The LSS is the Minkowski sum of a sphere and a line segment [6]. The advantage of this model lies in its simplicity. In particular, the location and orientation of each object is completely specified by the endpoints, P_A and P_B, and radius, r, of its LSS bounding volume. Thus, the obstacle avoidance problem can be reduced to enforcing a minimum distance threshold between a collection of continuous line segments.

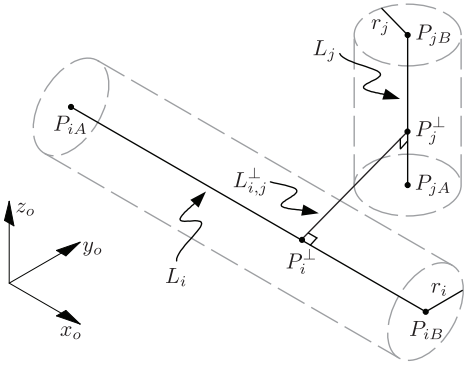

Fig. 1. Representation of robot links and workspace obstacles by line-swept spheres for collision avoidance computation. (Spherical end-caps not shown for clarity.)

Referring to Fig. 1, the distance between two LSS volumes, representing link i and obstacle j, can be evaluated using the parametric equations giving the loci of points lying on the underlying primitives

$$\begin{aligned} P_i &= P_{iA} + (P_{iB} - P_{iA})\gamma_i, \quad \gamma_i \in [0,1] \\ P_j &= P_{jA} + (P_{jB} - P_{jA})\gamma_j, \quad \gamma_j \in [0,1] \end{aligned} \tag{11}$$

The distance, $\delta_{i,j}$, between the two line segments for arbitrary values of the parameters is given by the 2-norm

$$\delta_{i,j} = \| P_{iA} + (P_{iB} - P_{iA})\gamma_i - P_{jA} - (P_{jB} - P_{jA})\gamma_j \| \tag{12}$$

Squaring (12) and grouping terms gives,

$$\begin{aligned} \delta_{i,j}^2 = A_{i,j}\gamma_i^2 + 2B_{i,j}\gamma_i - \\ 2C_{i,j}\gamma_i\gamma_j + 2D_{i,j}\gamma_j + E_{i,j}\gamma_j^2 + F_{i,j} \end{aligned} \tag{13}$$

where

$$\begin{aligned} A_{i,j} &= (P_{iB} - P_{iA}) \cdot (P_{iB} - P_{iA}) \\ B_{i,j} &= (P_{iB} - P_{iA}) \cdot P_{iA} - (P_{iB} - P_{iA}) \cdot P_{jA} \\ C_{i,j} &= (P_{iB} - P_{iA}) \cdot (P_{jB} - P_{jA}) \\ D_{i,j} &= (P_{jB} - P_{jA}) \cdot P_{jA} - (P_{jB} - P_{jA}) \cdot P_{iA} \\ E_{i,j} &= (P_{jB} - P_{jA}) \cdot (P_{jB} - P_{jA}) \\ F_{i,j} &= P_{iA} \cdot P_{iA} - 2(P_{iA} \cdot P_{jA}) + P_{jA} \cdot P_{jA} \end{aligned} \tag{14}$$

In (14), operator $(\cdot)$ refers to the vector dot product.

Using (13), the minimum value of $\delta_{i,j}^2$ can be found by constrained optimization. The minimization problem has the form

$$\begin{aligned} \text{minimize} \quad & Y(\gamma_i, \gamma_j) = A_{i,j}\gamma_i^2 + 2B_{i,j}\gamma_i - 2C_{i,j}\gamma_i\gamma_j + \\ & \qquad 2D_{i,j}\gamma_j + E_{i,j}\gamma_j^2 + F_{i,j} \\ \text{subject to} \quad & -\gamma_i \leq 0, \quad -\gamma_j \leq 0 \\ & \gamma_i - 1 \leq 0, \quad \gamma_j - 1 \leq 0 \end{aligned} \tag{15}$$

Problem (15) can be solved by invoking the Karush-Kuhn-Tucker (KKT) conditions [17]. In particular, the set of parameters, $\hat{\gamma}_i$ and $\hat{\gamma}_j$, that minimize $\delta_{i,j}^2$ are those for which the complementarity and stationarity conditions are satisfied. The Lagrangian of (15) is

$$\bar{Y}(\gamma_i, \gamma_j, \lambda) = Y(\gamma_i, \gamma_j) - \lambda_1^i\gamma_i + \lambda_2^i(\gamma_i - 1) - \lambda_1^j\gamma_j + \lambda_2^j(\gamma_j - 1) \tag{16}$$

Stationarity of the Lagrangian yields,

$$\begin{aligned} A_{i,j}\gamma_i + B_{i,j} - C_{i,j}\gamma_j + \tfrac{1}{2}(-\lambda_1^i + \lambda_2^i) = 0 \\ -C_{i,j}\gamma_i + D_{i,j} + E_{i,j}\gamma_j + \tfrac{1}{2}(-\lambda_1^j + \lambda_2^j) = 0 \end{aligned} \tag{17}$$

while the complementarity conditions give

$$\begin{aligned} -\lambda_1^i\gamma_i = 0, \quad \lambda_2^i(\gamma_i - 1) = 0 \\ -\lambda_1^j\gamma_j = 0, \quad \lambda_2^j(\gamma_j - 1) = 0 \end{aligned} \tag{18}$$

with $(\lambda_1^i, \lambda_2^i, \lambda_1^j, \lambda_2^j)^T \geq \mathbf{0}$.

The system of equations formed by (17) and (18) cannot be solved analytically. Therefore, we solve the values of $\hat{\gamma}_i$, $\hat{\gamma}_j$ and the Lagrange multipliers by integrating the above minimization problem as part of the manipulator optimal control problem (6). This is done by defining an augmented control vector, $\mathbf{u}^T = (\tau^T \mid \Delta_{i,j}^T \mid \ldots)$, where $\Delta_{i,j} = (\hat{\gamma}_i, \hat{\gamma}_j, \lambda_1^i, \lambda_2^i, \lambda_1^j, \lambda_2^j)^T$ is the vector of unknown parameters associated with problem (15). The associated stationarity and complementarity conditions are enforced by smooth path constraints that must be satisfied at each node.

By augmenting the control vector, the elements of vector $\Delta_{i,j}$ can be solved concurrently with the collision-free

link trajectories. In total, six additional dummy controls and six additional path constraints are introduced for each potential collision. Therefore, the proposed approach for obstacle avoidance scales proportionally to the number of obstacles. This is in contrast to other approaches for collision avoidance, e.g. [4], where the complexity of the optimization problem increases quadratically with the number of obstacles.

IV. Application Examples

This section presents the results of two different motion planning scenarios that demonstrate the efficacy of the proposed approach for collision-free trajectory optimization with stationary and dynamic obstacles. The trajectory optimizations were carried out using a commercially available implementation of the Legendre PS optimal control algorithm [19]. In each case, the feasibility of the numerical results were confirmed by executing the computed trajectories on experimental robotic manipulators (see Fig. 2). The robots are Cyton Alpha manipulators manufactured by Robai [20]. Only the base and shoulder joints of each robot were actuated in order to emulate a simple two-link manipulator. The dynamic model corresponding to this configuration is presented in [14].

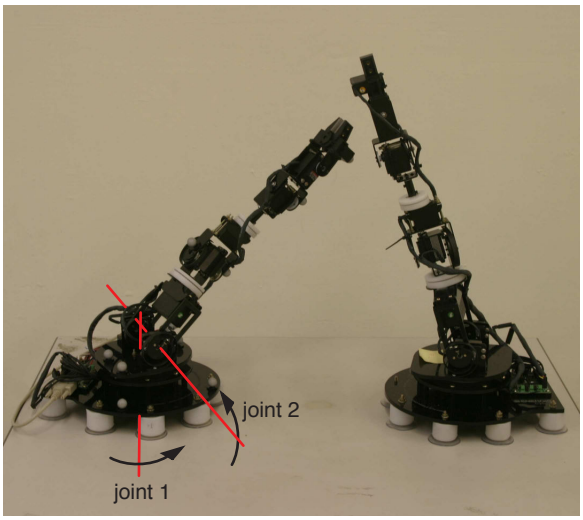

Fig. 2. Experimental robot manipulators.

Example 1: Static Obstacles

In this example, the objective is to complete a point-to-point maneuver while avoiding collisions with two stationary obstacles located in the workspace. The collision-free trajectory of the end-effector between initial and final positions, $Q^i = (19, 28, 48)$-cm and $Q^f = (2, -34, 48)$-cm, is shown in Fig. 3. It has been verified [21] that the trajectory satisfies the necessary conditions for optimality. Therefore, the collision-free trajectory is an extremal and a candidate global optimum. A plot of the separation distance between the robot and each obstacle is shown in Fig. 4. As is seen, the imposed constraint on the separation distance, $\delta_{\min} = 10$-cm, has been satisfied. The corresponding values of dummy control vector, $\Delta_{1,1}$, are shown in Fig. 5 for reference.

The time histories of the computed optimal position trajectories for each link of the robot are shown in Fig. 6 (solid lines). Referring to Fig. 6, the maneuver is completed in 1.16 sec. Fig. 6 also shows the experimentally measured position trajectories (dashed lines) that were obtained by executing the maneuver on the robot. Small discrepancies between the desired and actual joint profiles are observed due to practical nonidealities such as imperfect tracking characteristics of the joint servo motors, uncertainty in the robot inertial parameters, and unmodeled friction effects. Nonetheless, the strong consistency between the desired optimal trajectories and the experimental results confirms the feasibility of the numerical results.

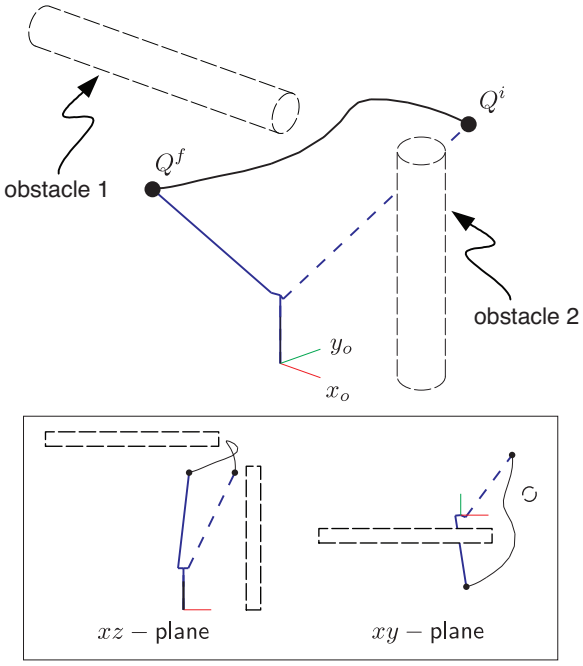

Fig. 3. Minimum-time maneuver in presence of stationary workspace obstacles.

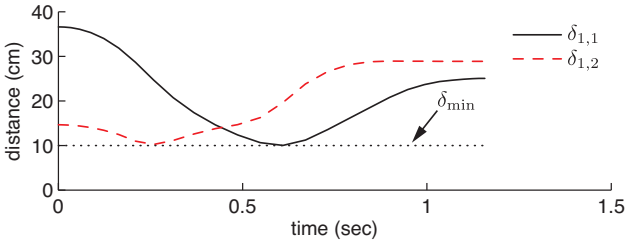

Fig. 4. Separation distance between robot and workspace obstacles.

Example 2: Dynamic Obstacles

This example involves the application of the proposed PS optimal control approach towards motion planning for

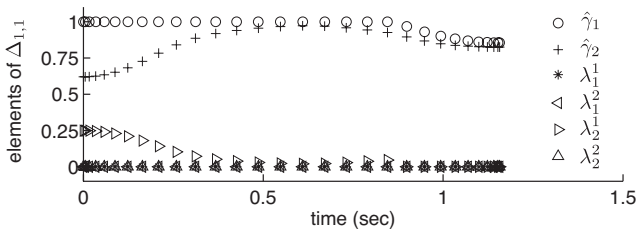

Fig. 5. Elements of vector $\Delta_{1,1}$ at the nodes.

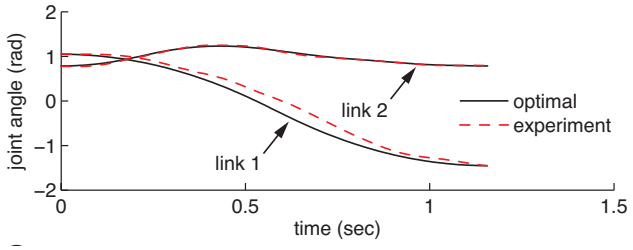

Fig. 6. Joint angles for maneuver in presence of stationary workspace obstacles. Legend: solid – computed optimal trajectory; dashed – experimental measurements.

a multi-robot system. The objective is to reposition the end-effector of robot 1 from $Q_1^i = (19, 28, 48)$-cm to $Q_1^f = (-19, -37, 38)$-cm and to move the end-effector of robot 2 from $Q_2^i = (15, -60, 48)$-cm to $Q_2^f = (-2, -6, 56)$-cm as quickly as possible, and without a collision. An acceptable distance, $\delta_{\min} = 7.5$ cm, should be maintained between the two robots throughout the maneuver.

One way of approaching this problem is to solve the optimal motion trajectories for each robot separately (in a decentralized fashion) and then implement each maneuver in sequence. This is an example of the time delay method for collision avoidance [22]. Fig. 7 shows the separation distances between the two robots corresponding to three different maneuver sequences: (*i*) both maneuvers executed simultaneously, (*ii*) robot 1 positioned first and then robot 2, and (*iii*) robot 2 positioned first followed by robot 1. Fig. 7 shows that it is only possible to complete the maneuver without violating the $\delta_{\min}$ constraint by repositioning robot 1 first followed by robot 2. The time delayed maneuver takes over 1.5 sec to complete.

A better way to solve the multi-robot motion planning problem is to generate the motion trajectories simultaneously, in a centralized manner. In this approach, the state and control vectors of each manipulator are stacked on top of one another. An advantage is that the optimization solver can consider the motion of each manipulator as a dynamic obstacle that the other should avoid. Therefore, the collision-free optimized trajectory for each robot can be executed simultaneously. This allows the maneuver time to be minimized while satisfying obstacle avoidance requirements. The optimal end-effector trajectories solved using the centralized PS optimal control approach are illustrated in Fig. 8, along with snapshots of the orientations of each robot arm at selected times during the maneuver. As before, the necessary conditions for optimality have been satisfied. Fig. 8 shows that the proposed algorithm has solved the trajectories of each manipulator in such a way that the multi-robot maneuver can be completed in 0.86 sec, while simultaneously satisfying the collision constraints. The separation distance between the two robots is reported in Fig. 9, which confirms that the required minimum distance threshold is maintained throughout the maneuver.

To verify the feasibility of the numerical results, the multi-robot maneuver was implemented experimentally on two robot manipulators. The robot position trajectories, shown in Fig. 10, confirm that the experimental hardware can track the desired optimal trajectories with a reasonable degree of accuracy. The experimental robots were able to carryout the maneuver without colliding, thus further substantiating the utility of the proposed PS optimal control approach for multi-robot trajectory planning.

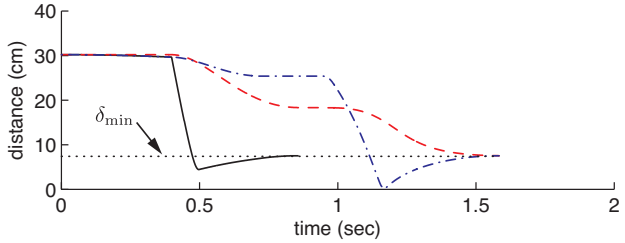

Fig. 7. Separation distance between robots for different movement sequences. Legend: solid – both maneuvers executed simultaneously; dashed – robot 1 positioned first; dash-dot – robot 2 positioned first.

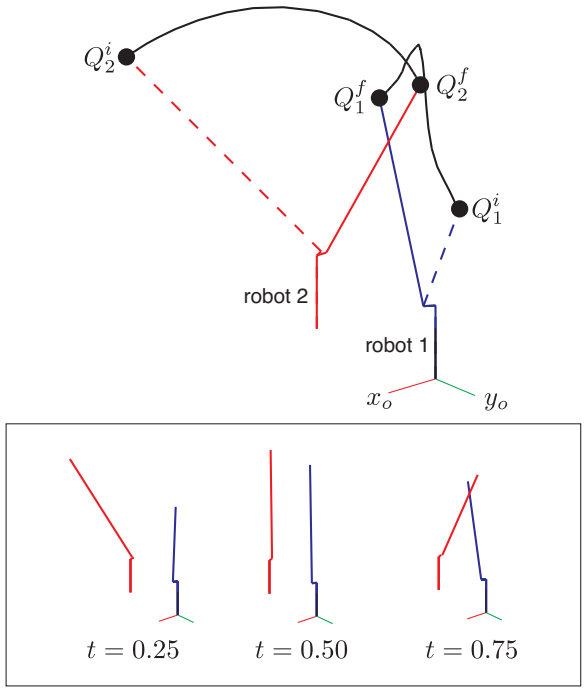

Fig. 8. Multi-robot maneuver solved using centralized PS optimal control approach.

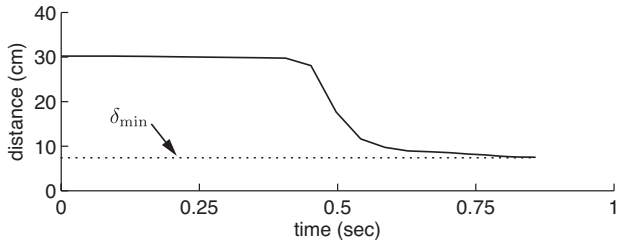

Fig. 9. Separation distance between robots using centralized PS optimal control approach.

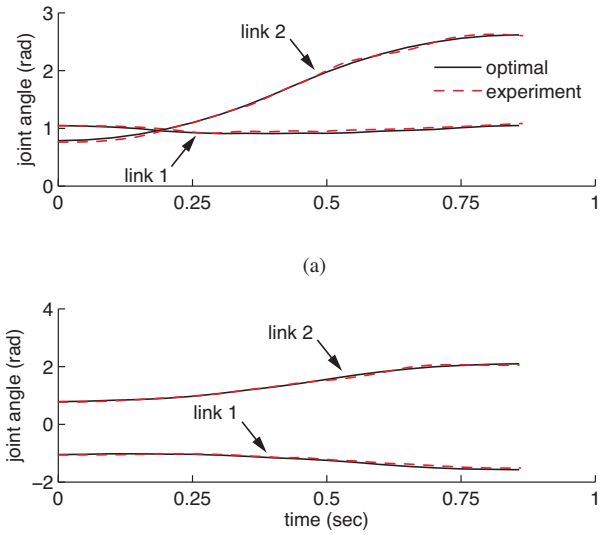

Fig. 10. Joint angles for multi-robot maneuver: (a) robot 1 joint angles; (b) robot 2 joint angles. Legend: solid – computed optimal trajectory; dashed – experimental measurements.

V. Conclusion

This paper presented a novel approach for robot manipulator trajectory planning in the presence of static and dynamic obstacles. A key feature of the approach was to incorporate the computation of inter-object separation distances into the manipulator optimal control problem as minimization subproblems through the use of KKT multipliers. By treating the KKT multipliers as augmented controls, it was possible to lift the dimension of the trajectory planning problem and solve it as a nonlinear PS optimal control problem with smooth constraints. Thus, non-smooth separation distance calculations, which are the crux of previous algorithmic approaches developed for proximity computation, could be avoided. The PS formulation also considered limits on the manipulator joint angles, velocities, accelerations, and joint torques to ensure the results are physically realizable. The practical utility of the proposed PS optimal control approach for collision-free manipulator trajectory planning was demonstrated on an experimental robot operating in the presence of several static workspace obstacles. A second experiment involving minimum-time centralized motion planning for a multi-robot system, in which each manipulator acts as a dynamic obstacle that the other should avoid, was also presented to further substantiate the approach.

References

[1] E. Gilbert, D. W. Johnston, and S. S. Keerthi, "A fast procedure for computing the distance between complex objects in three-dimensional space," *IEEE Transactions on Robotics and Automation*, vol. 4, no. 2, pp. 193–202, 1988.

[2] E. Larsen, S. Gottschalk, M. Lin, and D. Manocha, "Fast distance queries with rectangular swept sphere volumes," in *IEEE International Conference on Robotics and Automation*, vol. 4, 2000, pp. 3719–3726.

[3] S. F. P. Saramago and V. Steffen, "Optimal trajectory planning of robot manipulators in the presence of moving obstacles," *Mechanism and Machine Theory*, vol. 35, pp. 1079–1094, 2000.

[4] T. Chettibi, H. E. Lehtihet, M. Haddad, and S. Hanchi, "Minimum cost trajectory planning for industrial robots," *European Journal of Mechanics A: Solids*, vol. 23, pp. 703–715, 2004.

[5] R. R. dos Santos, V. Steffen, and S. Saramago, "Robot path planning in a constrained workspace by using optimal control techniques," *Multibody System Dynamics*, vol. 19, pp. 159–177, 2008.

[6] E. Larsen, S. Gottschalk, M. C. Lin, and D. Manocha, "Fast proximity queries with swept sphere volumes," Department of Computer Science, UNC Chapel Hill, Tech. Rep., 1999.

[7] C. G. Lo Bianco and A. Piazzi, "Minimum-time trajectory planning of mechanical manipulators under dynamic constraints," *International Journal of Control*, vol. 75, no. 13, pp. 967–980, 2002.

[8] M. Guilbert, L. Joly, and P.-B. Wieber, "Optimization of complex robot applications under real physical limitations," *International Journal of Robotics Research*, vol. 27, no. 5, pp. 629–644, 2008.

[9] B. J. Martin and J. E. Bobrow, "Minimum-effort motions for open-chain manipulators with task-dependnets end-effector constraints," *International Journal of Robotics Research*, vol. 18, no. 2, pp. 213–224, 1999.

[10] J. Elnagar, M. A. Kazemi, and M. Razzaghi, "The pseudospectral Legendre method for discretizing optimal control problems," *IEEE Transactions on Automatic Control*, vol. 40, no. 10, pp. 1793–1796, 1995.

[11] I. M. Ross and F. Fahroo, *New Trends in Nonlinear Dynamics and Control and their Applications*, ser. Lecture Notes in Control and Information Sciences. Springer-Verlag, 2003, vol. 295, ch. Legendre Pseudospectral Approximations of Optimal Control Problems, pp. 327–342.

[12] F. Fahroo and I. M. Ross, "Costate estimation by a legendre pseudospectral method," *Journal of Guidance, Control & Dynamics*, vol. 24, no. 2, pp. 270–277, 2001.

[13] Q. Gong, I. M. Ross, W. Kang, and F. Fahroo, "Connections between the covector mapping theorem and convergence of pseudospectral methods for optimal control," *Computational Optimization and Applications*, vol. 41, no. 41, pp. 307–335, 2008.

[14] J. Cascio, M. Karpenko, P. Sekhavat, and I. M. Ross, "Optimal collision-free trajectory planning for robotic manipulators: Simulation and experiments," in *Presented at 19th AAS/AIAA Space Flight Mechanics Meeting*, Savannah, GA, 2009.

[15] V. J. Lumelsky, "On fast computation of distance between line segments," *Information Processing Letters*, vol. 21, pp. 55–61, 1985.

[16] D. H. Eberly, *3D Game Engine Design: A Practical Approach to Real-time Computer Graphics*. San Francisco: Morgan Kaufmann, 2001.

[17] C. R. Bector, S. Chandra, and J. Dutta, *Principles of Optimization Theory*. Alpha Science International Ltd., 2005.

[18] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*. New York: John Wiley & Sons, 1989.

[19] I. M. Ross, *A Beginner's Guide to DIDO: A MATLAB Application Package for Solving Optimal Control Problems*, ser. Elissar Technical Report TR-711. www.elissar.biz, 2007.

[20] *Cyton Alpha 7D1G Operations Manual*. Philadelphia, PA: Robai, 2008.

[21] J. A. Cascio, "Optimal path planning for multi-arm, multi-link robotic manipulators," Master's thesis, Naval Postgraduate School, 2008.

[22] C. Chang, M. J. Chung, and B. H. Lee, "Collision avoidance of two general robot manipulators by minimum delay time," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 24, no. 3, pp. 517–522, 1994.