CONTROL OF A SERPENTINE ROBOT FOR INSPECTION TASKS

R.. Colbaugh* K. Glass* H. Seraji†

* Department of Mechanical Engineering, New Mexico State University
†Jet Propulsion Laboratory, California Institute of Technology

Abstract

This paper presents a simple and robust kinematic control scheme for the JPL serpentine robot system. The proposed strategy is den-eloped using the damped-] east-squares/confi.gura.tion control methodology, and permits the considerable dexterity of the JPL serpentine robot to be effectively utilized for maneuvering in the congested and uncertain workspaces often encountered in inspection tasks. Computer simulation results are given for the 20 degree-of-freedom (I) OF) manipulator system obtained by mounting the twelve DOF serpentine robot at the end-effector of an eight DOF Robotics Research arm/lathe-bed system. These simulations demonstrate that the proposed approach provides an effective method of controlling this complex system.

1. introduction

Space platforms, such as Space Station Freedom, arc complex and expensive scientific facilities which must operate for extended periods in harsh and remote environments. The inspection of such platforms, leading to preventive maintenance as opposed to breakdown repair, is therefore an important task. The Jet Propulsion Laboratory (J PL) has been involved in a NASA-funded research and development project on Remote Surface Inspection since October 1990 [I]. The objective of this project is to develop and demonstrate the core technologies that arc needed for remote inspection of space structures using a robot manipulator carrying the inspection sensors at its cnd-effecter. The purpose of the inspection is to monitor the health and observe the status of the space structure, and to take the necessary corrective actions when anomalies are detected. Complete inspection of the trusses and other structural components that make up a space platform, as well as the equipment which is mount ed on it, requires the capability to maneuver safely in congested and uncertain workspaces and to reach into truss openings and other constricted environments. An essential component of this inspection system is a dexterous robotic arm for sensor placement and the associated control system. The robot used in this project is an anthropomorphic seven-jointed arm from the Robotics Research Corporation (RRC) mounted on a mobile platform with translational motion capability. 'l'he "extra." degree-of-freedom (D OF) possessed by this arm provides high dexterity and versatility that is not available in conventional six DOF arms. For example, the arm redundancy has been used to control the elbow position, to provide collision avoidance in reaching through an opening, and to avoid joint limits [e.g., 2-4].

It can be seen that the ability to perform automated collision-free motions in a congested environment is a fundamental requirement for robotic surface inspection. While the seven DOF RRC arm is highly dexterous, it is physically not capable of reaching into small cavities or operating in a cluttered environment. In order to provide such functional capabilities, JP L has developed a "serpentine" appendage that can be mounted at the end-effecter of the RRC arm (see Figure 1). By virtue of its high dexterity and flexibility, the serpentine robot has the physical capability to reach into and maneuver inside hard-to-access workspaces and perform inspection or m anipulat ion tasks su ccessfull y. The development of the serpentine robot was initiated at JPL in 1993. A six-jointed twelve DOF serpentine robot was designed and fabricated at J PL. When the serpentine is appended to the RRC arm, the total system will have 20 DOF and will be highly redundant. While this 20 DOF robotic system certainly has the potential to maneuver in congested environments without colliding with workspace obstacles, realizing this potential requires that this complex robot be effectively controlled.

This paper considers the problem of controlling the JPL serpentine robot system for the execution of remote inspection tasks. The proposed approach to maneuvering the serpentine robot in congested and uncertain workspaces is to decompose the motion control problem into two subproblems: first, find a collision-free path for the tip of the serpentine robot, and then control the links of the robot so that they '(slither" after the tip. Wc define slithering to be the motion obtained by causing each of the joints to track the same task-space path as the tip of the robot. The problem of finding a collision-free path for the tip of the serpentine robot can be solved using any of a number of available approaches to sensor-based collision avoidance for (small) mobile robots; for example, a popular approach is the artificial potential field method [e.g., 5]. Given this tip path, the joint motions required to obtain the desired slithering motion of the arm can be computed at the differential inverse kinemati c level using the damped-least-squares implement ation of configurate ion control [4]; implicit in this approach is the assumption that a joint-space position controller is available. The simplicity and computational efficiency of the proposed approach allows the scheme to be implemented for real-time control during remote inspection tasks.

2. Kinematic Control Scheme

In this section, the kinematics of the serpentine robot system is briefly characterized, and a control scheme is proposed that permits this robotic system to safely maneuver in obstacle fields without *a priori* knowledge of the workspace geometry. The JPL serpentine robot system consists of a six-jointed twelve DOF serpentine robot attached to the end-effector of a seven D OF RRC arm, which is in turn mounted on a mobile platform possessing one DO F. The kinematics of the resulting 20 DOF system may be characterized qualitatively by noting that the serpentine robot joints provide a sequence of relative pitch and yaw motion between adjacent links, the RRC arm joints permit relative roll and pitch motions between links, and the mobile platform gives a translational motion capability to the entire system. .4 quantitative description of the robot kinematics is provided in Figure 2, in which the DH parameters for the robot are listed.

'I'he proposed control strategy for the serpentine robot is now presented. An underlying feature of the scheme is the use of proximity sensors, mounted at the tip of the serpentine robot, to detect the presence of workspace obstacles. The utilization of such sensors for achieving collision-free motion with (small) mobile robots is well-understood [e. g., 5]. The collision avoidance problem for robotic arms is considerably more difficult, however, because it involves simultaneously controlling the motion of several physically interconnected links in a collision-free manner. The proposed approach to maneuvering the serpentine robot in congested and uncertain workspaces is to decompose the motion control problem into two subproblems: first, find a collision-free path for the tip of the serpentine robot using the sensors at the tip and one of the many available mobile robot algorithms, and then *control* the links of the robot so that they "slither" after the tip. We define slithering to be the motion obtained by causing each of the joints to track the same task-space path as the tip of the robot while remaining one link length apart; the analogy with the slithering of an actual snake is clear. Note that because each pair of adjacent links in the JPL serpentine robot arc connected by a two DOF' joint that provides relative pitch and yaw motion between the links, it is (kinematically) possible to obtain this desired slithering motion. It can be seen that a slithering motion will ensure collision-free maneuvering in a static obstacle field, and is particularly well-suited for moving through small openings or maneuvering in a very congested workspace.

As indicated above, the determination of a collision-flee path for the tip of the serpentine robot using proximity sensor information can be achieved by treating the robot tip as a mobile robot and implementing one of the available mobile robot obstacle avoidance schemes (e.g., the potential field approach). Given this tip path, the joint motions required to obtain the desired slithering motion of the arm can be computed using the following kinematic control strategy (recall it is assumed that the robot has low-level joint servo-loops capable of ensuring that the robot joint-space position 0 $\in \Re^{20}$ remains close to any desired trajectory $\theta_d(t)$). Let x $\in \Re^3$ denote the position of the robot tip, and note that the forward kinematics and differential kinematics for the manipulator can be written as

$$\mathrm{x} = \mathbf{f}(\theta) \quad , \qquad \dot{\mathrm{x}} = J_e(\theta)\dot{\theta} \tag{1}$$

where f : $\Re^{20} \rightarrow \Re^3$ and $J_e \in \Re^{3\times 20}$ is the end-effector Jacobian matrix. Our approach to kinematic control is to utilize *the improved configuration control method* described in [4]. Briefly, this method involves augmenting the robot task-space with additional user-specified tasks and then inverting the augmented differential kinematics using a damped-least-squares (DLS) solution. Thus the user defines the kinematic functions y = g(6), where $g : \Re^{20} \rightarrow \Re^{r}$ and r is the number of kinematic functions to be controlled as the additional task. Augmenting (1) with these kinematic functions yields the kinematic and differential kinematic models

$$\mathbf{z} = \begin{bmatrix} \mathrm{x} \\ -- \\ \mathrm{y} \end{bmatrix} = \begin{bmatrix} \mathrm{f}(\theta) \\ -- \\ \mathrm{g}(\theta) \end{bmatrix} \tag{2}$$

$$\dot{\mathbf{z}} = \begin{bmatrix} J_e(\theta) \\ --- \\ J_c(\theta) \end{bmatrix} \dot{\theta} = J(\theta)\dot{\theta} \tag{3}$$

where z $\in \Re^{(3+r)}$ is the configuration vector and $J_c \in \Re^{r \times n}$ is the Jacobian matrix associated with the additional task, Given the desired evolution of the configuration vector $\mathbf{z}_d(t) = [\mathbf{x}_d^T(t) \mid \mathbf{y}_d^T(t)]^{T}$, the DLS approach to determining the desired joint-space trajectory $\theta_d(t)$ corresponding to $\mathbf{z}_d$ is to compute $\dot{\theta}_d$ as [4]

$$\dot{\theta}_d = [J^T W J + \lambda^2 W_v]^{-1} J^T W [\dot{\mathbf{z}}_d + Ke] \tag{4}$$

and then integrate $\dot{\theta}_d$ to obtain θ_d. In (4), $\mathbf{e} = \mathbf{z}_d$ - z, $K \in \Re^{(3+r)\times(3+r)}$ is a symmetric positive-definite feedback gain that corrects for linearization error and other unmodeled effects, and the relative importance of closc trajectory tracking and low joint velocities is reflected in the choice of the symmetric positive-definite weighting matrices $W \in \Re^{(3+r)\times(3+r)}$ and $W_v \in \Re^{20 ,20}$ and the positive scalar constant λ. The D LS formulation of the configuration control scheme gives approximate solutions to (3) that arc robust to singularities. This is achieved by optimally reducing the joint velocities to induce minimal errors in the task performance by modifying the task trajectories.

The desired behavior of the serpentine robot can now be specified by defining the desired configuration vector $\mathbf{z}_d(t) = [\mathbf{x}_d^T(t) \mid \mathbf{y}_d^T(t)]^T$. The desired tip trajectory $\mathbf{x}_d(t)$ "is obtained by treating the tip as a mobile robot and determining a collision-free inspection path using a mobile robot path-finding strategy [e.g., 5.]. Given this tip path, the desired trajectory $\mathbf{y}_d(t)$ for the additional task vector is specified to ensure that the desired slithering motion of the arm is realized. More specifically, the requirement that the six (two DOF) joints of the serpentine robot are to track the same path as the robot tip is expressed as a set of twelve kinematic constraints. Thus we define $\mathbf{g} = [\mathbf{y}_1^T \mid \mathbf{y}_2^T \mid \cdots \mid \mathbf{y}_6^T]^T \in \Re^{12}$, where $\mathrm{y}_i \in \Re^2$ defines the location of joint i of the serpentine robot (indexed from the tip) relative to the manipulator base frame (note that only two coordinates are required to completely locate each joint because of the physical constraint that joint $i+1$ must remain one link length from joint i). The desired evolution of this additional task $\mathbf{y}_d(t)$ is then constructed recursively by requiring that each of the joints follow the desired robot tip path $x_d(t)$ while remaining onc link length apart. This can bc conveniently accomplished by recording any two (of the three) coordinates of the tip path $x_d(t)$(paramctcrizcd by arc length measured along the path) on-line, and then defining each of the $y_{di}(t)$ accordingly. The details of this approach to achieving the desired slithering motion can be found in [6].

3. Simulation Example

The kinematic control strategy dcscribcd in Section 2 is now applied to the 20 DOF JPL serpentine manipulator in a computer simulation example. The simulation environment consists of a Personal IRIS Graphics Workstation, the Hydra graphics rendering package, and the software necessary to realize the DLS configuration control implementation of the slithering strategy. Figure 1 is a graphical depiction of the JPL surface inspection testbed, including the complete robotic system and truss structure. The objective of the simulation example is to illustrate the increased dexterity obtained by mounting the 12 DOF serpentine manipulator as the end-cffector of the 8 DOF RRC arm/mobile platform robotic system. In this examplc, the cnd-cffcctor is commanded to track a trajectory that rcachcs inside one of the triangular-openings in the truss, moves vertically on the intcrior of thc truss, and then moves out the top of the truss. A collision-free trajectory is generated using a strategy similar to the one given in [5], and the links of the serpcntinc robot are controlled to slither after the end-cffector as described in Section 2. The truss location and geometry, temporal trajectory of the end-cffector, controller weights, and a more detailed discussion of the implcmcntation of the slithering algorithm are given in [6]. Figure 1 shows the final manipulator configurate ion and confirms that the slithering control of the scrpentine robot allows collision-free motion in a cluttered environment.

4. Conclusions

This paper presents an efficient and robust kinematic control schcme for the JPL serpentine robot system. The proposed strategy is developed using the damped-least-squares/configuration control methodology, and provides the capability to maneuver in the congested and uncertain workspaces often encountered in inspection tasks. Computer simulation rcsults demonstrate that the proposed approach provides an effective method of controlling the highly redundant robotic system.

It is noted that, whilc the collision avoidance strategy proposed in this paper is useful for maneuvering in static obstacle fields, there are important applications that require thc robotic system to safcly ncgotiatc dynamically-varying environments. The proposed strategy can bc modified to permit collision avoidance in a dynamic obstacle field by using the mcthods presented in [7,8]. Briefly, the idea is to mount proximity sensors on all of the links, compress the proximity sensor information regarding the location of the (dynamically-varying) obstacles into a set of kinematic incquality constraints using the approach in [7], and then ensure that these inequality constraints are satisfied (and therefore the obstacles are avoided) using the method given in [3,8]. The development and implementation of this strategy is a subject of current research.

5. Acknowledgements

The research described in this paper was carried out, in part, at the Jet Propulsion Laboratory under contract with the National Aeronautics and Space Adminstration. Additional support for the work was provided by the Department of Energy (WERC) and the Army Research Office.

6. References

1. Hayati, S., J. Balaram, H. Seraji, W. Kim, and K. Tso, "Remote Surface Inspection System", *Robotics and Autonomous Systems*, Vol. 11, No. 1, 1993, pp. 45-59
2. Seraji, H., "Configuration Control of Redundant Manipulators: Theory and Implementation", *IEEE Transactions on Robotics and Automation,* Vol. 5, No. *4, 1989,* pp. *472-490*
3. Colbaugh, R., H. Seraji, and K. Glass, "Obstacle Avoidance for Redundant Robots Using Configuration Control", *Journal of Robotic Systems,* Vol. 6, No. 6, 1989, pp. 721-744
4. Seraji, H. and R. Colbaugh, "Improved Configuration Control for Redundant Robots" , *Journal of Robotic Systems,* Vol. 7, No. 6, 1990, pp. 897-928
5. Khatib, O., "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots" , *International Journal of Robotics Research*, Vol. 5, No. 1, 1986, pp. 90-98.
6. Colbaugh, R., K. Glass, and H. Seraji, "A Slithering Algorithm for the JPL Serpentine Robot", Robotics Laboratory Report 93-01, New Mexico State University, June 1993
7. Boddy, C. and J. Taylor, "Whole-Arm Reactive Collision Avoidance Control of Cinematically Redundant Manipulators", *Proc. 1993 IEEE International Conference on Robotics and Automation,* Atlanta, GA, April 1993
8. Glass, K., R. Colbaugh, D. Lim, and H. Seraji, "On-line Collision Avoidance for Redundant Manipulators", *Proc. 1993 IEEE International Conference on Robotics and Automation,* Atlanta, GA, April 1993

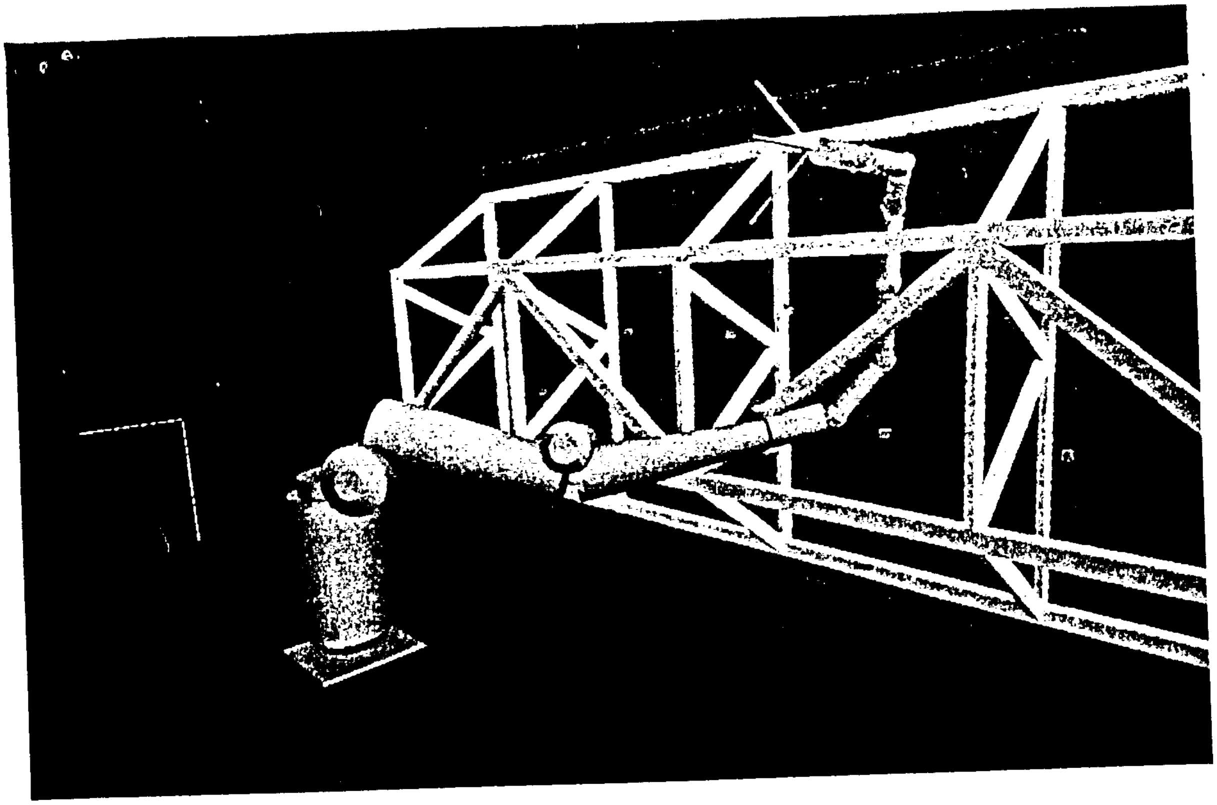

Figure 1: Graphical representation of Serpentine Robot

i	1	2	3	4	5	6	7	8	9	10
α_{i-1}	0	π/2	π/2	π/2	π/2	π/2	π/2	π/2	-π/2	-π/2
a_{i-1}	0	Θ	12.32	10.79	7.94	7.94	4.92	4.92	0	0
d_i	0	0	0	54.6	0	54.6	0	20.0	0	0
θ_i	0	π/2	π	π	π	π	π	0	-π/2	0

i	11	12	13	14	15	16	17	18	19	20	21
α_{i-1}	π/2'2	–π/2	π/2/2	-π/2	π/2	-π/2	π/2	-π/2	π/2	-π/2	θ
a_{i-1}	220.0.(	0	2020.0	0	20.0	0	20.0	0	20.0	0	20.0
d_i	0	0	0	0	0	0	0	0	0	0	0
θ_i	0 00	0 0 0 (0)0	[illegible]	[illegible]	0	0	0	0	0	0	0

Figure 2: DH Parameters for Serpentine Robot