MECH TOC

Proceedings of DETC'02
ASME 2002 Design Engineering Technical Conferences
and Computer and Information in Engineering Conference
Montreal, Canada, September 29-October 2, 2002

DETC2002/MECH-34336

PLANNING OF SAFE KINEMATIC TRAJECTORIES FOR FREE FLYING ROBOTS APPROACHING AN UNCONTROLLED SPINNING SATELLITE

Stephen Jacobsen
Field and Space Robotics Laboratory
Massachusetts Institute of Technology

Christopher Lee
Field and Space Robotics Laboratory
Massachusetts Institute of Technology

Chi Zhu
Field and Space Robotics Laboratory
Massachusetts Institute of Technology

Steven Dubowsky
Field and Space Robotics Laboratory
Massachusetts Institute of Technology

ABSTRACT

The problem of planning a safe trajectory for a free-flying robot to approach an uncontrolled spinning satellite is addressed. First, a heuristic plan is presented for a simple planar case, which constructs a collision-free path within realistic system constraints. Second, a general numerical optimization technique for planning a safe spatial trajectory is presented. In it, the approach path is parameterized and a cost function based on performance metrics is minimized in order to find the optimal path. The results are analyzed, and it is shown that optimization techniques can be used to produce a far safer approach trajectory than the heuristic method.

1. Introduction

Robotic servicing of satellites, including rescue, repair, refueling, and maintenance, promises to extend satellite life and reduce costs, making it one of the most attractive areas of developing space technology [1, 2]. The capture and retrieval or repair of an expensive satellite that has lost attitude control is an important class of future missions for space robots (Fig.1).

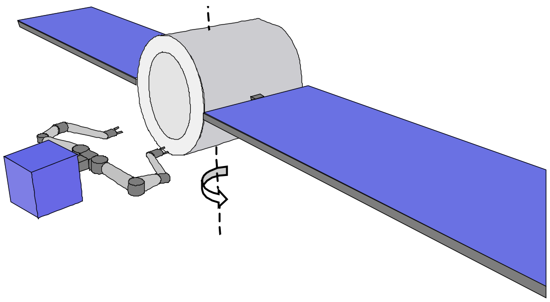


Figure 1. Capture of a spinning satellite by a space robot

Nearly all of the satellite servicing missions that have occurred have been carried out by astronauts and, in some instances, with limited help of robotic manipulators. The use of astronauts to capture uncontrolled spinning satellites is both expensive and dangerous, even for slowly spinning satellites. Thus, it is desirable to develop free-flying space robots that are capable of carrying out such satellite servicing missions.

The high value of the satellites that are potential targets for servicing and the considerable investment of time and money in a robotic servicing mission make the safety of the robot and satellite a top priority. The major danger is the potential for collision between the satellite and the robot. For example, collision with the panels of a spinning satellite could seriously damage sensitive electronics or delicate appendages and render the robot and satellite inoperable, so the chance of such collisions must be minimized. This paper presents a method for planning the safest kinematic trajectory during this approach phase of capture. It is shown that optimization methods can be used to produce an approach trajectory that is safer, as defined by a safety metric, than a heuristically planned trajectory.

2. Background and Literature

Several on-orbit servicing tasks have been performed by astronauts with some assistance from robotics systems. Examples include NASA Missions STS-61, STS-82, and STS-103, in which the Hubble Space Telescope was repaired by astronauts with the help of the Remote Manipulator System (RMS, or "Canadarm") and Mission STS-49, the capture and redeployment INTELSAT-VI, stranded in unusable orbit due to launch failure. The INTELSAT-VI mission showed the difficulty and danger of using astronaut EVAs to capture and stabilize even a very slowly spinning satellite. Hence, free-flying robotic solutions to the satellite capture and service problem are very attractive in such cases.

There has been a great deal of fundamental research in the area of space robotics, including work on dynamics and control, navigation, and sensor systems [3, 4]. Methods for control of space robotic systems with complex dynamics have been developed [5]. Work has included methods to reduce the angular momentum of a spinning satellite by a series of carefully planned mechanical impulses in preparation for capture [6]. In 1997, NASDA's ETS-VII satellite successfully demonstrated the rendezvous and docking (RVD) and space robotics (RBT) technologies for a cooperative target satellite [7]. However, the problem of maximizing the safety in the capture of an uncontrolled satellite remains largely an unaddressed problem.

Optimization-based path planning for fixed-base industrial robots has been addressed for a number of years. Methods have been developed to simultaneously optimize geometric path of a robot and its speed along the path [8]. In this paper, both the geometric path and the speed along the path are combined into a single parameterized representation of a trajectory that is optimized directly. Researchers have demonstrated an optimal control approach for planning the trajectory of a space robot derived from Pontryagin's maximum principal, but the robot model and cost function are very simple, and obstacle avoidance is not addressed [9]. Others have used a randomized motion planning method to perform online obstacle avoidance for robots with dynamic constraints in environments with many moving obstacles [10]. This work is successful for quickly finding good paths for avoiding obstacles in complex environments, but the trajectory planning method presented here exploits the predictability of stable motion in the space environment.

This method addresses the problem of planning the safest possible kinematic trajectory for a space robot to approach an uncontrolled satellite in preparation for grasp and stabilization. Two methods are presented for planning a trajectory. The first is a heuristic path plan for the special case where the robot motion is constrained to a plane perpendicular to the spin axis of the satellite (i.e. the "2D case"). This case provides insights into the fundamental issues of the problem. Second, a more general method for kinematic trajectory planning using numerical optimization methods is presented. It is applied to the representative 2D case as well as the general 3D case where the robot is not restricted to remain within a plane.

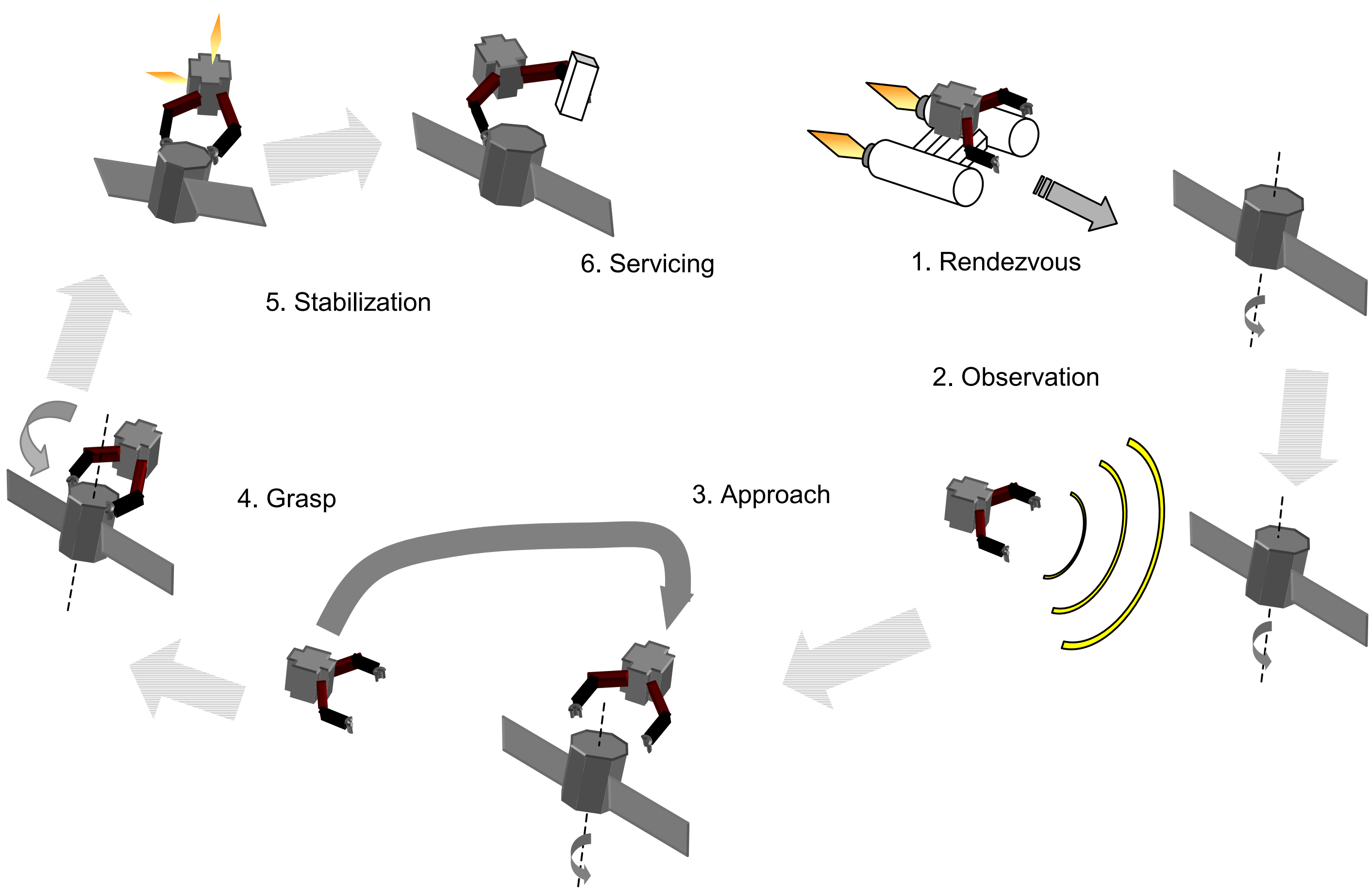


Figure 2 - Phases of a Satellite Capture Mission

3. The Approach Problem

A mission scenario for satellite capture and servicing by a free-flying robot is illustrated in Fig. 2. This mission can be divided into a sequence of separate tasks: rendezvous, observation and data acquisition, approach, grasp, stabilization, and satellite retrieval or servicing. Safe approach and capture of the satellite are both critically important and technically challenging parts of the process [11].

In the approach phase, the robot starts at a position *A* that is 100-500m away from the target satellite, at the origin of reference frame *R* (Fig. 3). The satellite is located at (x_s, y_s) and is spinning around its center *s* with ω_s; the radii of satellite body and satellite panel are r_s and r_p, respectively. At the end of the approach phase, the robot is orbits about the satellite with radius r_c. Its end-effectors can then be extended to the grasp point x_g. The danger of a collision between robot and satellite exists when the robot is within the panel radius, r_p, of the center of the satellite (region S_p).

A set of representative parameters for a nominal case are given in Table 1.

Table 1. Representative parameters for satellite and space robot

(x_s, y_s)	(50, 100) m	ω_s	3 rpm
r_s	2 m	m_r, mass of robot	100 kg
r_p	30 m	m_s, mass of satellite	5000 kg
r_c	3 m	Isp, specific impulse	200 sec
Maximum thruster force	50 N	Approach time	1 – 5 min.

In this work, it is assumed that:

1. Satellite and robot safety is the highest priority, namely, collision between the satellite or its panels and robot should be avoided, even in the case of a system failure.
2. The satellite is not specifically designed for capture by the robot, and the location of the grasp point depends on existing hard points on the satellite body.
3. The satellite has lost attitude control and is in a stable spin.
4. The space robot is equipped with two manipulator arms, an attitude control system, and a fuel supply, which is a limited and expensive resource.
5. Orbital mechanics and influences such as gravity gradient torque and atmospheric drag are not significant during the short approach phase.

4. The Planar Case

To identify and study the fundamental issues of the approach problem, a case in which a robot is constrained to move in the plane perpendicular to the axis of rotation of the satellite is first studied. Several paths from position A to the position were considered, including a simple straight line path and a blended straight-line/spiral path.

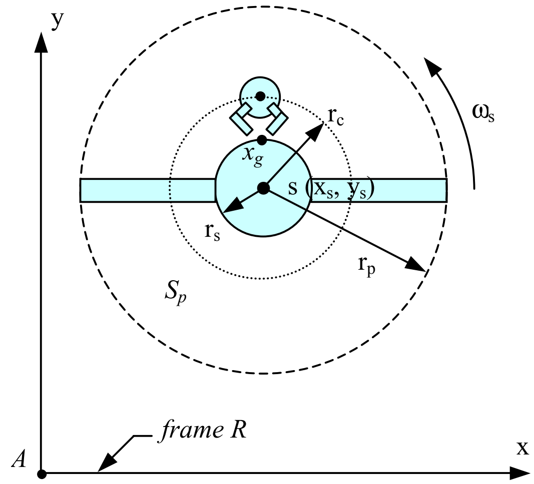


Figure 3. Satellite with robot in grasp configuration

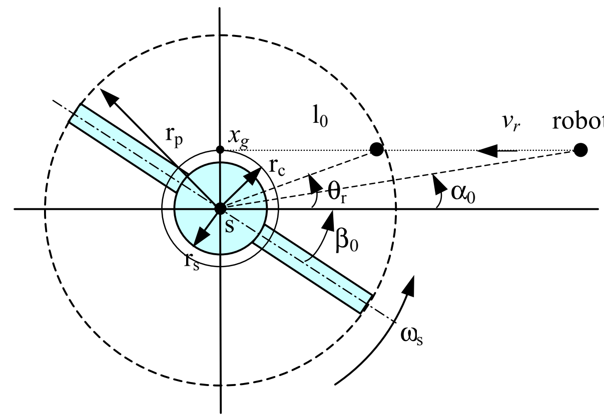


Figure 4. A straight line path

a. *Straight line path*. On this path, the robot enters into the orbit r_c along a straight line with constant velocity $v_r = r_s\omega_s$ as shown in Fig. 4. To guarantee that the robot will not collide with the satellite or its panels, it can be shown that:

$$\theta_s < \theta_r \tag{1}$$

where $\theta_s = \omega_s t - \beta_0$ is the angle of the satellite with respect to the vector v_r, $\theta_r = \tan^{-1}(r_c/(l_0 - v_r t))$, and β_0 is the initial angle of the satellite panel that is always less than π. Thus,

$$\frac{l_0}{r_c} < \frac{\pi}{2} + \beta_0 < \frac{3\pi}{2} \tag{2}$$

For most satellites, the solar panel radius is rather large, and the condition of inequality (2) cannot be satisfied, so a simple straight-line path is not acceptable.

b. Blended straight-line/spiral path. Adding a spiral section to the path overcomes the problem of collision in the straight line (Fig. 5). In it, the robot follows a straight line path to a distance r_b from the satellite, then spirals in to the grasp configuration. The velocity of the robot along the path during approach is shown in Fig. 6.

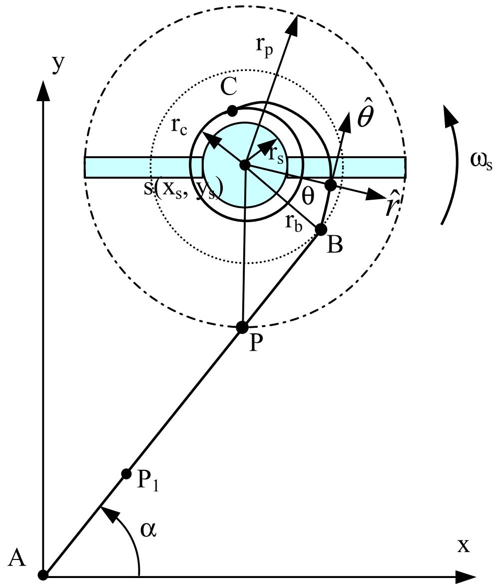


Figure 5. Blended Straight-line/spiral Path

The procedure to determine the path is as follows:

1. Find curve *BC*: For simplicity, the curve *BC* is assumed to be a spiral with ω_s, given by:

$$\begin{aligned} x(t) &= x_s + r(t)\cos(\omega_s(t-t_B)-\theta) \\ y(t) &= y_s + r(t)\sin(\omega_s(t-t_B)-\theta) \end{aligned} \quad (3)$$

where t_B is the time from start point *A* to point *B*; θ is the start angle of the spiral at point B; $r(t)$ is the radius of the curve from the satellite center *s*. $r(t)$ can be determined from initial condition and the condition that the path should be smooth.

2. Find point *B*. After the spiral *BC* is determined, point *B* can be found by the condition that straight line AP_1PB is tangential to point *B*. The force that moves the robot along the spiral *BC* should be less than the maximum thruster force. From this constraint, the time from *B* to *C* can be determined. For nominal case, the total time is $t_{total} = 53.5$ sec; and the total fuel usage is $\Delta m_{fuel} = 0.986$ kg when the given thruster force is 50 N.

These results show that, for a reasonable set of system parameters, an approach can be completed in less than one minute using a reasonable thruster force and modest fuel consumption. Hence, safety becomes the critical issue during the approach phase. By changing the departure time from *A* with respect to the angular position of the satellite, it is possible to avoid a collision, and the configuration of the robot on the orbit at r_c can also be adjusted by changing t_{BC} and departure time. However, for the planar case, the robot spends a substantial amount of the total time in the region between the rotating satellite panels. A thruster interruption, sensor error, or computing problem could result in a collision and mission failure.

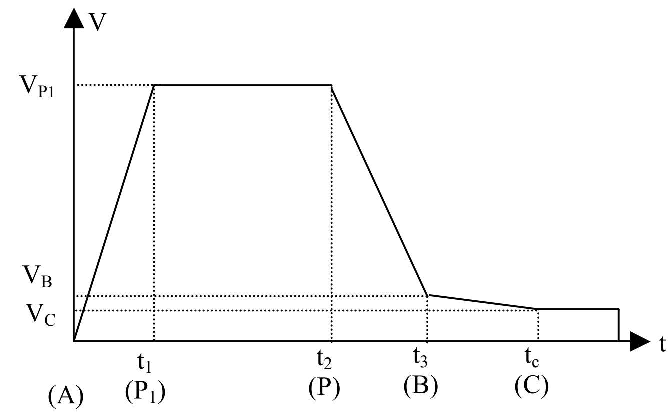


Figure 6. Velocity change during approach

5. Path Planning Using Optimization Methods

To enhance the system robustness to collision, the approach path planning problem was treated as an optimization problem. Optimization based path planning consists of three fundamental steps. First, the path is expressed in terms of parameters that describe its kinematic shape and velocity profile along the path. Second, a cost function is defined, which evaluates the quality of the path for each set of parameters based on performance metrics and applies a penalty for any undesirable attributes. Finally, an optimization routine is used to determine the values of the path parameters that yield the lowest total cost.

The performance metrics used in the cost function include the important measures of performance during satellite approach; safety, fuel usage, time required, and suitability of the final configuration for grasp and stabilization. Physical system constrains, such as maximum thruster force, can also be represented in the cost function. As shown by the planar analysis, reasonable approach times and fuel consumption within a realistic maximum thruster force can be readily achieved. However, maximizing safety remains a critical concern.

Expressing the safety of a path in mathematical terms is a less obvious process than conventional quantitative performance metrics, such as fuel usage or time. Possible safety metrics include ones that maximize distance from the satellite during approach or minimize time within an area close to the satellite, and each will produce a different choice of optimal path.

Metrics used for the cost function in this study are:

1. Safety metric based on time to collision: proportional to $1/t_i$, where t_i is the time to impact with the satellite in the event of loss of thruster power, based on the current trajectory (Fig. 7)
2. Fuel usage: estimated based on mass of robot, acceleration, and Isp of propellant
3. Time to rendezvous: proportional to total approach time squared
4. Acceleration constraint: keeps acceleration of robot within a_{max} limited by maximum thruster force

The trajectory of the robot is specified as

$$path = f(\mathbf{v}) \quad (4)$$

where f is the form of path representation, and $\mathbf{v}$ is the vector of path parameters. In its most general form, the parameterization specifies both the trajectory of the robot body and the robot configuration (including arm joints) at each point in time. This paper will address a subset of this general parameterization; the translational position and velocity is accounted for by the path parameterization, but not the angular position or configuration of the robot.

The cost function for the optimization is a path integral

$$P(\mathbf{v}) = \int_{t_0}^{t_f} g(path_v(t))\,dt + h(path_v) + l(path_v(t_f)) \qquad (5)$$

based on performance metrics *g* (e.g. fuel usage, safety, time), with additional terms for costs *h* related to the path as a whole (e.g., maximum thruster force required), and costs *l* related to the grasp configuration (e.g., manipulability of grasp configuration). The cost should be a smooth function of the parameterization vector.

The path is parameterized using polynomic splines with sufficient degrees of freedom on each spline section to match boundary conditions. Subject to boundary conditions of position and velocity, cubic splines or parabolic splines with a single cubic section, the latter used in calculations for this paper, are able to meet these requirements. Waypoints on each spline section combine with the boundary conditions to fully define the path.

For the case presented here, the robot is treated essentially as a point mass as it moves along the path in order to simplify evaluation of the cost function at each point. In the more generalized case, robot configuration and end-state boundary conditions can be parameterized to allow variation in the final grasp velocity and configuration.

If the cost function used by the optimization routine contains local minima within the parameter space and the optimization routine does not rigorously search for the globally optimum solution, then supplying different initial parameter values to the routine may result in different paths being chosen as optimal. These different solutions may be equally desirable if they are caused by the existence of symmetry and yield the same value when evaluated by the cost function. Separate locally optimal solutions may, however, differ drastically from one another in their total cost, and care must be taken in the selection of initial parameter values when working with a cost function known to possess local minima. For this reason, a heuristically determined acceptable path is a good choice as a starting point for an optimization routine, if such a path is available.

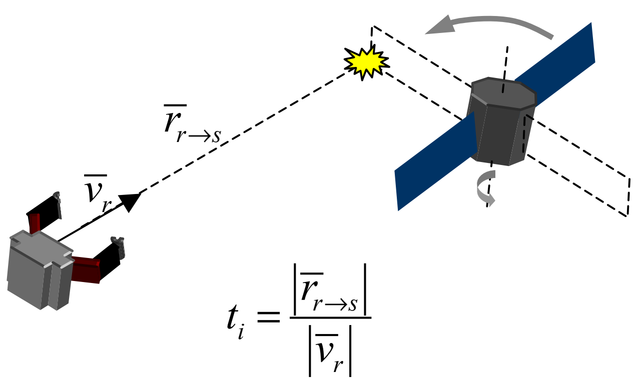


Figure 7. Safety metric based on time to collision

By perturbing the set of parameters $\mathbf{v}$ and evaluating $P_i(\mathbf{v}_i)$, the algorithm moves $\mathbf{v}_i$ in the direction that produces the largest decrease in P and iteratively finds the parameter set $\mathbf{v}^*$ which minimizes P

$$\mathbf{v}^* = \min_{\mathbf{v}}(P(\mathbf{v})) \qquad (6)$$

In this study, a numerical approach to optimization is adopted for flexibility, and because the general cost function is not suitable for closed-form solution. The Nelder-Mead [14] minimization algorithm is used in this case because it does not require explicit representation of the gradient of $P(\mathbf{v})$.

When the path has been optimized for a given set of *n* waypoints, *n* is incremented and a new $P(\mathbf{v}^*)$ is found. This process is repeated while the addition of waypoints significantly improves the total cost of the path. When $\Delta P_{n\to n+1}$ falls below a specified tolerance, the stopping criteria is met optimization of the path is complete.

6. Results

6.1 2D optimization results

For the representative 2D scenario, the cost function was discretized for numerical optimization as

$$P = k_t t_r^{\,2} + \sum_1^m P_i \Delta t_i \qquad (7)$$

where

$$P_i = k_s s_p + k_f f_p + k_a a_p$$

m = number of evaluation points on path
t_r = total time for approach

and s_p, f_p, and a_p are the metrics for safety, fuel usage, and acceleration, respectively. The acceleration metric limits the acceleration to values less than $a_{max} = 0.5\ m/s^2$ based on a maximum thruster force of 50 N and robot mass of $m_r = 100\ kg$. The constants k_s, k_f, and k_a are weighting factors within the cost function. The values for system parameters (x_s, y_s, m_r, etc.) used in the cost function are the same as those used in the heuristic case.

A seed path defined by 2 waypoints, shown in Figure 8, was first optimized with respect to the shape of the path, with the time intervals on each spline section held constant.

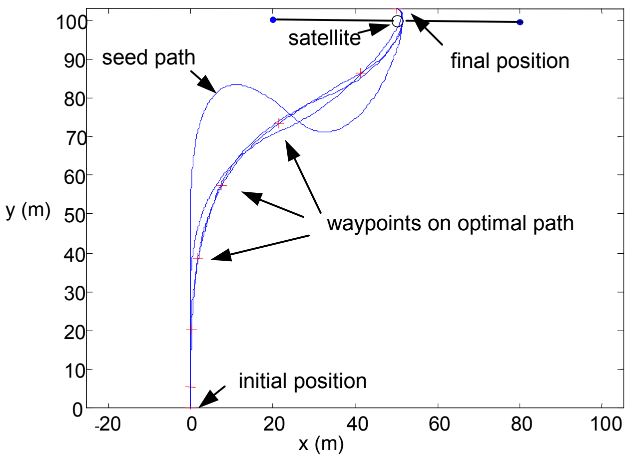
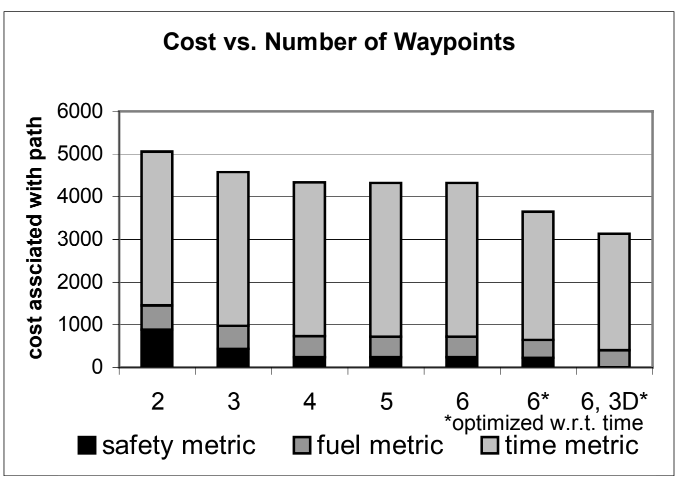


Figure 8. 2D Path resulting from optimization

Waypoints were added and the optimization routine repeated, until the total cost of path began to converge, shown in Fig. 9. The $n = 6$ path was then used as a seed path for the optimization routine, and the time interval on each spline section was allowed to vary, further lowering the total cost of the path. The absolute acceleration profile for the optimal 2D path is shown in Fig. 10. The metrics used in this case cause the robot to accelerate most aggressively near the end of the approach, when the robot is in close proximity to the satellite.



6.2 3D optimization results

Using the results of 2D optimization as a seed path, the routine was repeated for the 3D case by allowing waypoints to vary in the z direction as well. The safety metric was modified to take the position and velocity of the robot in the z direction into account when checking the possibility of collision and calculating time to impact. As shown by the cost distribution for the 3D path in Fig. 9 and the 3D plot, shown in Fig. 11, the cost associated with the safety metric is reduced. This is because the robot travels above the panels of the rotating satellite and drops down to rendezvous near the end of the path, and there is no potential for collision along most of the path.

Fuel usage is also reduced because the total length of the optimal 3D path is less than that of the 2D path, while acceleration is not greatly increased, as Fig. 12 indicates.

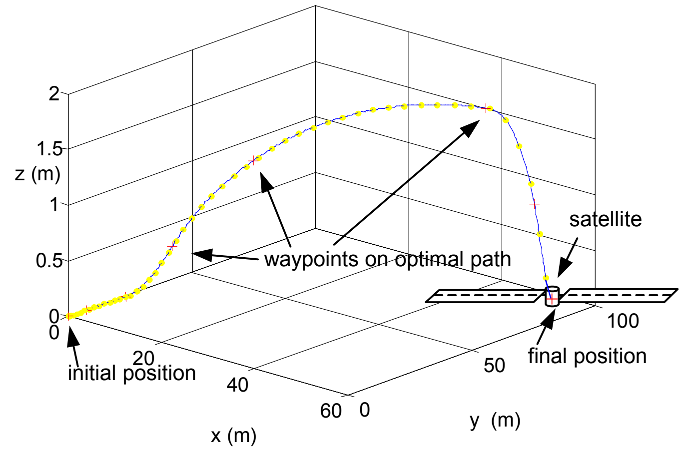


Figure 11. 3D Path produced by optimization

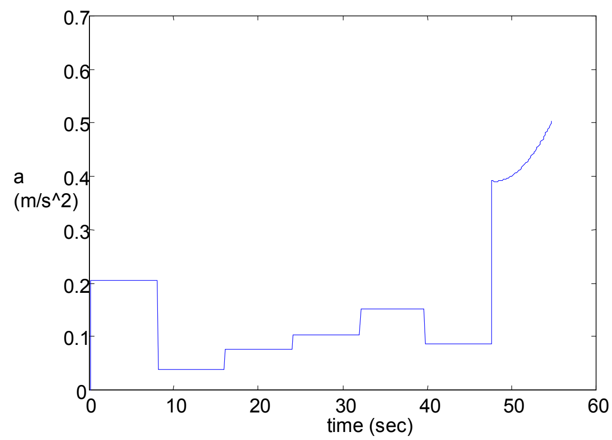


Figure 10. Acceleration profile for 2D path

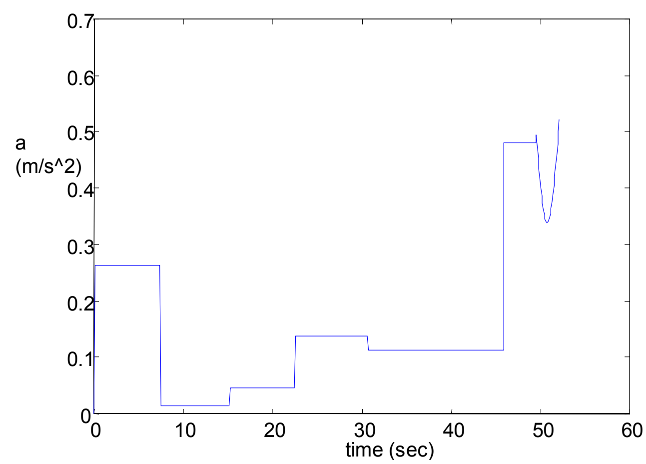


Figure 12. Acceleration profile for 3D path

6.3 Comparison with heuristic path planning method

After performing optimization for the 2D and 3D cases, the analytically determined 2D path was evaluated using the same cost function. Table 2 summarizes the resulting data. Time does not vary greatly for the different cases, but the cost based on the time to collision safety metric is significantly higher for the heuristic path. This metric is designed to minimize the time to collision integrated along the path, and is a somewhat subjective valuation of the safety of a path. Small changes in a path can produce large variations in the value of the safety metric, so this metric provides a means for relative comparison rather than an absolute measure of safety.

Table 2. Summary of Path Data

	t_r (sec)	fuel (kg)	total cost	safety metric	fuel metric	time metric
2D Heuristically Determined	53.5	0.986	1421600	1417800	986	2862
2D Optimized	54.8	0.42	3650	224	420	3005
3D Optimized	52.2	0.408	3129	≈ 0	407	2722

7. Conclusion

This paper presented a numerical optimization method for planning the approach trajectory of a robot catching an uncontrolled spinning satellite. The proposed planning method consists of parameterizing possible trajectories via a spline representation, then numerically optimizing the trajectory against a cost function. The resulting paths are far safer, based on the safety metric, than a path designed using a heuristic method. By optimizing the kinematic shape and velocity along the path, an approach trajectory was created that is robust to failure in the attitude control system of the space robot. The robot reaches the required position and velocity boundary conditions, but its velocity at any point on the path will not result in a collision with the satellite or its panels.

The cost function is the key element in determining the outcome of the optimization process, and the measure of a path's relative safety path is the most important component of the cost function. In this paper, the measure of safety for a given point along a trajectory was the time to collision with the satellite from that point if the robot's thrusters were to fail. This worked well, but introduced local minima into the cost function, such that the chosen path was not necessarily the globally optimal path. Other measures of safety might be more appropriate for a given mission. The measured safety of the path was the most significant difference between the heuristic and the optimized paths. Fuel usage and the total time to complete the approach are similar for both methods, and are realistic for the nominal mission scenario.

Acknowledgements

The authors would like to acknowledge NASDA for their support of this work. Stephen Jacobsen is supported by a National Science Foundation Graduate Research Fellowship.

References

1. King, D., 2001, "Space Servicing: Past, Present and Future", Proceedings of the 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space: i-SAIRAS 2001, Montreal, Canada.
2. Yoshida, K., 2000, "Space Robot Dynamics and Control: To Orbit, From Orbit, and Future", Robotics Research, The Ninth International Symposium, Eds, Hollerbach, J.M, and Koditschek, D.E, pp.449-456, Springer.
3. Xu, Y. and Kanade, T., November 1992, "Space Robotics: Dynamics and Control" Kluwer Academic Publishers, ISBN 0-7923-9265-5.
4. Ullman, M. A. March 1993 "Experiments in Autonomous Navigation and Control of Multi-Manipulator Free-Flying Space Robots" PhD thesis, Stanford University, Stanford,CA.
5. Dubowsky, S. and Papadopoulos, E., October 1993, "The Kinematics, Dynamics, and Control of Free-flying and Free-floating Space Robotic Systems" IEEE Transactions on Robotics and Automation 9 (5): 531-543.
6. Kawatmoto, S., Matsumoto, K., and Wakabayashi, S., 2001, "Ground Experiment of Mechanical Impulse Method for Uncontrollable Satellite Capturing", Proceedings of the 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space: i-SAIRAS 2001, Montreal, Canada.
7. Oda, M., April 2000, "Experiences and lessons learned from the ETS-VII robot satellite", Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA 2000), San Francisco, CA.
8. Shiller, Z., and Dubowsky, S., 1991, "On computing the global time-optimal motions of robotic manipulators in the presence of obstacles", IEEE Transactions on Robotics and Automation, 7(6): pp. 785-797.
9. Sakawa, Y., 1999, "Trajectory planning of a free-flying robot by using the optimal control. Optimal Control Applications and Methods, 20, 235-248.
10. Kindel, R., Hsu, D., Latombe, J., and Rock, S., April 2000, "Kinodynamic motion planning amidst moving obstacles" IEEE International Conference on Robotics and Automation.
11. Matsumoto et. al., May, 2002, "Satellite Capturing Strategy Using Agile Orbital Servicing Vehicle, Hyper-OSV", IEEE International Conference on Robotics and Automation 2002, Washington, D.C.
12. Wertz, J. R., 1985, "Spacecraft Attitude Determination and Control", D. Reidel Publishing Company.
13. Sidi, M. J., 1997, "Spacecraft Dynamics and Control – A Practical Engineering Approach", Cambridge University Press.
14. Nelder, J. A. and Mead, R., 1965, "A Simplex Method for Function Minimization." Comput. J. 7, 308-313.

