AIAA Infotech@Aerospace 2007 Conference and Exhibit, Rohnert Park, CA, May 7-10, 2007.

Pseudospectral Optimal Control: A Clear Road for Autonomous Intelligent Path Planning

Kevin P. Bollino[*], L. Ryan Lewis[†], Pooya Sekhavat[‡], and I. Michael Ross[§]

Guidance, Navigation, and Control Lab
Department of Mechanical and Astronautical Engineering
Naval Postgraduate School, Monterey, CA, 93943

This paper captures the essence of intelligent path planning by posing the problem in a framework based on optimal control theory. Design requirements for autonomous vehicles call for on-board intelligence capable of making timely decisions, performing tasks in a "smarter" fashion, and ultimately accomplishing missions with extreme accuracy. This is the definition of an optimal control problem! Indeed, most, if not all, motion planning problems can be formulated and solved using optimal control techniques. Motivated by the significant advancements in optimal control techniques over the last decade, we demonstrate the broad range of unmanned systems that can operate both optimally and autonomously by solving path-planning problems using pseudospectral methods.

I. Introduction

One of the common desires in the advancement of technology is a way to make systems smarter. Developing a more intelligent system provides more efficiency and effectiveness with less external intervention – reliable automation. The revolution of personal computers was a major step towards the growing trend of removing humans out of the loop and integrating intelligence and cognitive science into the systems. With current advances in miniaturization, high-end computing, and extremely efficient software development, intelligent systems is now a key buzz word throughout the general engineering community, particularly in such areas as automation and control. Many engineering initiatives are focused on the development of unmanned systems that have the capability to operate with minimum to no human interaction. For example, the Jet Propulsion Laboratory (JPL) and NASA Ames Research Center are conducting a joint effort on autonomous spacecraft using artificial intelligence (AI) to automate the control of on-orbit vehicles.[1] Primarily chartered by the Office of the Secretary of Defense (OSD), the government is taking a leading role in developing the architecture for the domain of unmanned systems. Such efforts were fueled by the establishment of the Joint Robotics Program (JRP) in 1989[2] where the concept of network-integrated unmanned systems is used for Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) applications. Under development by the Unmanned Systems Branch of SPAWAR Systems Center, San Diego (SSC San Diego), such "multi-dimensional ISR" is Integrated Force Protection whereby unmanned ground vehicles (UGVs), unmanned aerial vehicles (UAVs), and unmanned surface vehicles (USVs) work together as force protection units in the battlefield.[3] Figure 1 shows an example of such an unmanned ensemble for a search and rescue mission.

The idea of autonomous operations has been the long-term goal of robotics for decades. Since the inception of robotics in the early 1960's, the basic idea was to sense the state of tasks, adapt to changes, and select appropriate actions needed to accomplish a goal.[4] In terms of modern control theory, this is essentially navigation/estimation, adaptive/optimal control, and goal-oriented control. There are various elements that contribute to making a system autonomous (control design/planning, execution, system ID, sensor fusion, etc) and each element is constrained by three fundamental factors: mission complexity, environmental difficulty, and human interface. The real "brain" of any autonomous system is the element of control design/planning. Planning is one of the primary elements of human intelligence that makes us extraordinary. In addition to our acute sensory perception, data processing, and

[*] Postdoctoral Research Associate, AIAA Member, Email: bollinok@verizon.net.
[†] Graduate Student
[‡] Research Scientist, AIAA Member, Email: psekhava@nps.edu.
[§] Professor, AIAA Associate Fellow, Email: imross@nps.edu.

motor control skills, most of our deliberate actions (i.e. motions) are a result of our plans. Whether they are carefully and consciously carried out or rapidly and subconsciously reacted, it is the intellectual capability of planning that allows us to accomplish a goal. Thus, to enable unmanned autonomous vehicles to effectively carry out various missions, real-time path planning is paramount.

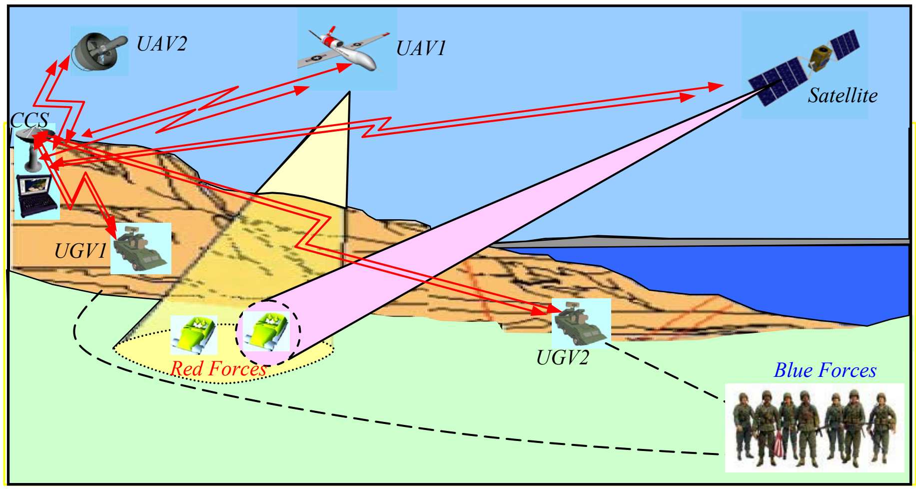


Figure 1. Rendition of Cooperative Multi-Unmanned System.

Planning and scheduling is based on the ability of a system to generate low-level sequence of tasks that can accomplish formalized high-level goals. Traditional planning and scheduling relies on humans to pre-program a set of actions to accomplish a pre-planned task or series of tasks. The objective in designing an intelligent system is to automatically determine the tasks required to accomplish the overall mission. Figure 2 illustrates various Guidance, Navigation, and Control (GNC) methods that attempt to satisfy certain requirements and capabilities that are associated with an intelligent/autonomous system.

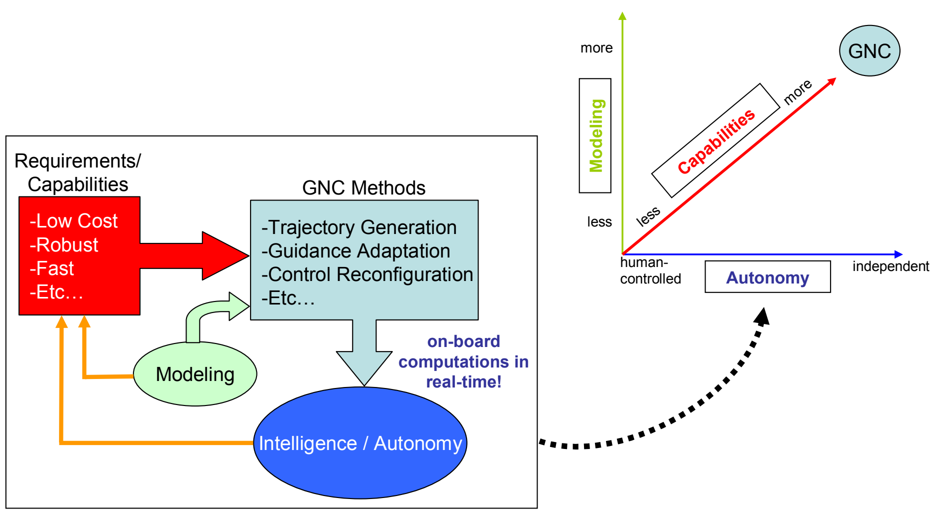


Figure 2. Using GNC Methods for Intelligence/Autonomy.

Note that modeling, in terms of both vehicle and environment, plays a key role in how GNC methods provide enhanced capabilities. In other words, it doesn't matter how autonomous the system is, if it operates based on a very crude model, then its effectiveness (i.e. capability) will be significantly hindered.

The purpose of this paper is to demonstrate how optimal control theory can be applied to various unmanned systems in order to provide them with the intelligence needed to take the desired action. Figure 3, adopted from the Joint Architecture for Unmanned Systems[5], shows the current autonomy spectrum for various unmanned system (e.g. robots) objectives. Note that the location of "path planning" is not very high and right on the chart. By implementing the optimal path planning technique described in this paper, our examples demonstrate the ability to shift this objective further towards 100 % autonomy/intelligence.

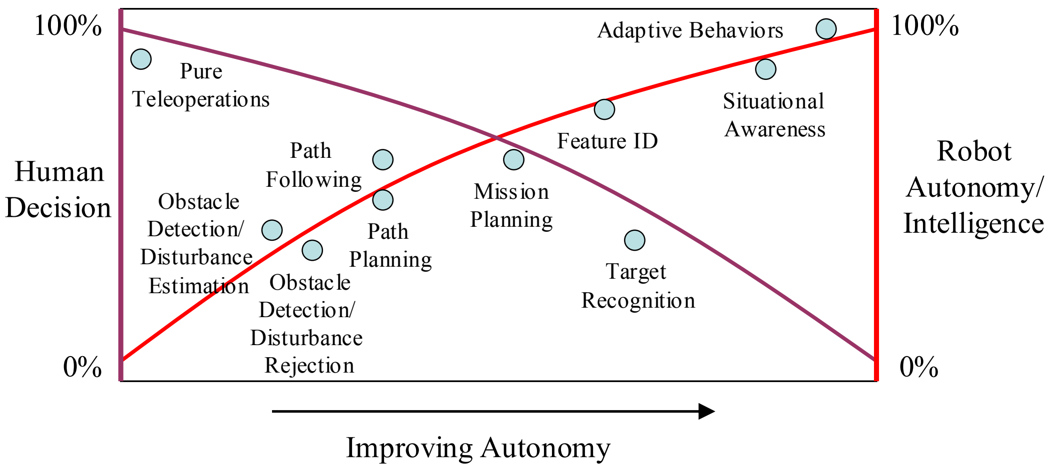


Figure 3. Autonomy Spectrum Adapted from JAUS[5].

In the next section, we will first briefly cover some of the traditional approaches for autonomous path planning. In section III, we will introduce our optimal, real-time path planning scheme. Finally, section IV will present the application of the method on four example vehicles illustrated in Fig. 4.

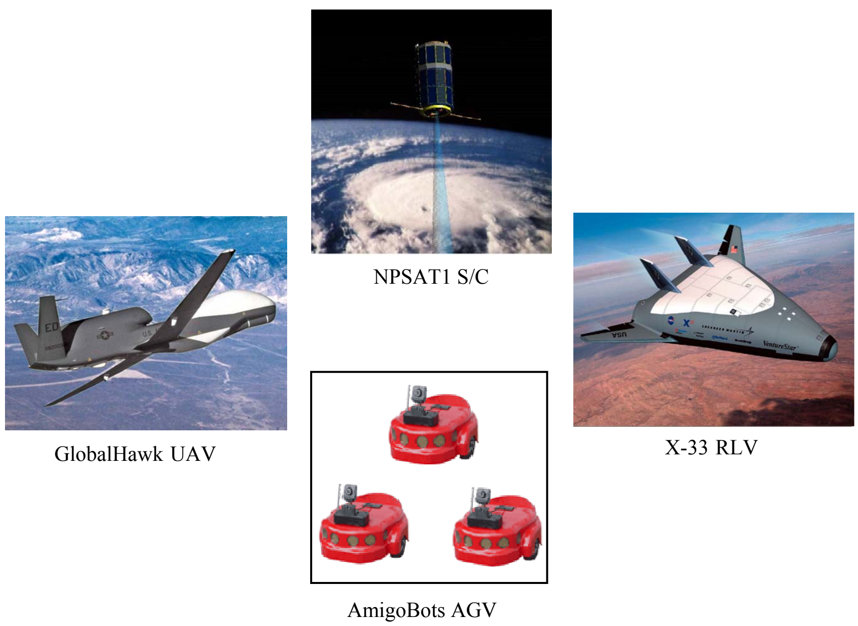


Figure 4. Pictures/Schematics of Unmanned Vehicles Used to Demonstrate the Scheme.

II. Traditional Approaches to Autonomy

Machine autonomy is, traditionally, a discipline fertilized by the robotics and computer science fields. In this regard, vehicle motion planning and control is no different. Most methods that are being pursued today rely on potential functions, sampling methods and probabilities, heuristics, and pre-programmed behaviors. These traditional approaches have disadvantaged the autonomous systems in three areas. Firstly, engineers are forced to separate the path planning and the tracking problems. Secondly, optimality is sacrificed for the purpose of feasible solution generation. Lastly, the system complexity is unnecessarily simplified to fit the previous molds, thereby representing the problem at hand inappropriately. As explained in the subsequent sections, optimal control methods present a drastic divergence from the previous schools of thought and open a new avenue toward autonomy. First, we briefly review some of the traditional approaches towards autonomy.

A. Artificial Intelligence (AI)

AI has historically been associated with robotics software engineering and has been separately employed for various parts of the overall mission such as planning, commanding action, perceiving, learning, etc. Today, the focus of the research and development activities is on combining such efforts into an integrated AI system capable of processing multiple components simultaneously, more like the human mind. One way to achieve this goal is through the use of artificial neural networks.

Pioneered by Bernard Widrow of Stanford University in the 1950's[6], the concept of Artificial Neural Networks (ANNs) was adopted from neuro-biological sciences. As defined by Webopedia[7], a neural network is

> *A type of artificial intelligence that attempts to imitate the way a human brain works. Rather than using a digital model, in which all computations manipulate zeros and ones, a neural network works by creating connections between processing elements, the computer equivalent of neurons. The organization and weights of the connections determine the output.*

In the realm of mobile robotics, ANNs are primarily used to generate the control signals directly from sensor data inputs. An example of this approach is the Carnegie-Mellon's ALVINN project – an autonomous land vehicle that uses a feed-forward network to take an input image of the road and directly transform it into a steering direction.[8]

The premise of ANNs is a supervised learning technique called back-propagation, whereby a correct result is provided for a series of training scenarios (i.e. given an input, here is the answer). With weights associated with each "neuron", output errors are propagated backwards such that each weight is modified. This incrementally improves the error. To enhance the learning process, genetic algorithms can be integrated into the process such that the learning takes place based on "staged evolution." Naturally, ANNs require a time-consuming period of training the network in addition to the long execution time. Also, until recently, ANNs did not have the capability to adapt to the substantial changes in the data.

B. Potential Functions

Introduction of potential functions dates back more than thirty years. The method's conception was under the auspices of limited computational requirements and the need to react under changing environments. An intuitive visualization of this method is to imagine the vehicle to be a positively charged particle in the state space. Obstacles have a similar positive charge – repulsive force – and the target location has a negative charge – attractive force. The accumulation of these forces drives the vehicle towards the goal.

The method of utilizing potential functions benefits from simplicity and functionality. The potential function needs only to be real valued and differentiable over the state space. Inherently, the method provides a continuous control input at every position in the state space; therefore, path tracking and feedback are unnecessary. The limitations of the method lie in the solution optimality, local minima, and difficulty in managing vehicle constraints. Significant research directed toward the local minima problem has led to developments that can mitigate or eliminate the problem.[9-11] Meanwhile, the optimality and vehicle constraint problems remain intractable for a technique which derives the control inputs solely from sensory measurements.[12]

C. Sampling Methods

Great enthusiasm has been shown towards the application of sampling methods to the trajectory planning and dynamic planning problems. Within ten years of the method's development, such considerable excitement has thrust this technique to the forefront of the application tools. The force behind this movement is the ease with which the method handles high dimensionality and complex vehicle and global constraints. Similar to potential function methods, sampling methods are not based on a rigorous mathematical structure. Despite the existence of numerous distinct sampling techniques (e.g. PRM, RRT, EST, SRT, SBL), they all share similar defining actions. In general, sampling techniques synthesize a dense tree or roadmap of nodes and edges. Each node represents a particular instantaneous state of the vehicle and an edge connects the two nodes via a planned path. Construction of the trees or roadmaps is through random or otherwise generation of new nodes and then connecting the new nodes to the old ones via a planned path that can incorporate the vehicle constraints. Once the construction phase is completed (i.e. a connection exists between start and finish), post processing on the resulting route is typically necessary and desired to ensure smoothness and improve optimality.

Overall, these methods are excellent at quickly generating feasible solutions; however, the solutions realized are by no means optimal.[13] Furthermore, complex paths and increased dimension of the system kinematic and dynamic equations can "bog down" the computational speed.[14]

III. The Optimal Control Framework for Autonomous Intelligence

In the early days of computer technology and before advancements in robust computational techniques, large-scale optimization computations were so time-consuming that it was almost impractical to calculate the optimal solutions. Today, advancements in sparse linear algebra, development of new algorithms, and improved computer processor speeds have made solving optimization problems relatively easy and fast. This renders optimal control theory a more viable technique in the development of intelligent systems. In fact, recent applications of real-time optimal control[15-19] have proven to be very promising in facilitating feedback solutions to complex nonlinear systems. For more complex intelligent systems, feedback can include a multitude of sensor inputs in the process (sensor fusion). The final example in this paper is the culmination of this concept, but first, we briefly introduce the approach.

A. Optimal Control Problem

The general idea of optimal control is to generate an optimum control trajectory that drives the system from an initial condition to final states while minimizing (or maximizing) a performance index (cost function), subject to constraints. The cost function is typically represented in a Bolza form,

$$J\left(\mathbf{x}(\cdot),\mathbf{u}(\cdot),t_0,t_f\right) = E\left(\mathbf{x}(t_0),\mathbf{x}(t_f),t_0,t_f\right) + \int_{t_0}^{t_f} F\left(\mathbf{x}(t),\mathbf{u}(t),t\right)dt \tag{1}$$

where the Mayer term, $E(\cdot)$, is considered a fixed cost and the Lagrange term, $F(\cdot)$, is a running cost over time. The cost can be minimized with respect to the states, **x**, the controls, **u**, and/or the clock times, $\mathbf{t_0}$ and $\mathbf{t_f}$, subject to the constraint equations that include nonlinear dynamics, paths constraints, endpoint conditions, and limits on states and controls.[20-23] Each of the intelligent autonomous path planning examples presented in this paper is, first, formulated into the above standard optimal control problem definition.

B. Pseudospectral Method

To solve the nonlinear optimal control problem, a spectral algorithm known as the Legendre Pseudospectral (PS) method is employed. The method, available through a MATLAB-based software package called DIDO[24], discretizes the problem and approximates the states, co-states and control variables using Lagrange interpolating polynomials where the unknown coefficient values are obtained at the Legendre-Gauss-Lobatto (LGL) node points. The embedded nonlinear programming (NLP) solver SNOPT[25], which is based on sequential quadratic programming, then solves a sequence of finite-dimensional optimization problems that capture the full nonlinearities of the system. Unlike other traditional methods, PS methods uses efficient (i.e. Gaussian) discretization and sparsity to transform large-scale optimization problems into a sequence of significantly smaller-scale problems; hence improving both speed and convergence properties. The method has been successfully used for solving a large number of open-loop optimal control problems and was recently extended into a PS-feedback control method that provides closed-loop optimal solutions capable of managing uncertainty.[18] The following examples illustrate how optimal control techniques can be utilized to add intelligence to unmanned vehicles.

IV. Examples of Intelligent Path Planning

A. Intelligent Trajectory Generation for a Reentry Vehicle

This sub-section presents intelligent reentry guidance for an X-33-class reusable launch vehicle (RLV). The objective is to optimally plan a path from the reentry point (suborbital in this case) to the landing site or some designated endpoint in the vicinity thereof. The complexity of this mission is mainly driven by the drastically changing environment that the vehicle undergoes while transcending the atmosphere. Of primary concern are the heating rate, the dynamic pressure, and the structural loads exerted on the vehicle as a result of hypersonic speeds through large altitude-density variations. The following example demonstrates the proposed method's ability to rapidly re-generate feasible and optimal trajectories to various landing sites in Florida. Historically, as in the case of Space Shuttle guidance, it takes many man-hours of pre-flight planning to design such trajectories off-line and then pre-program the Shuttle's flight computer with numerous waypoints and contingency trajectories. Despite being one of the most astonishing designs in aerospace history, the Space Shuttle lacks the autonomous capability to generate new trajectories "on-the-fly." This limitation creates a potentially lethal safety risk and as such, has fueled tremendous efforts to improve the guidance capabilities of future RLVs. Here, we demonstrate how optimal trajectory generation injects elements of intelligence into the now autonomous path planning.

As described in Ref. 26, we use the Final Approach Corridor (FAC) as the means to automatically generate the necessary endpoint conditions (e.g. altitude, speed, and attitude) required for proper alignment with the intended runway. The performance index, J, is simply the missed distance to the center of a prescribed cross-section of the FAC. Therefore, the optimal control problem is to find the optimal path that minimizes the missed distance represented by the following cost function

$$J(\mathbf{x}(t)) = \left(h_{FAC} - h_f\right)^2 + \left(\lambda_{FAC} - \lambda_f\right)^2 + \left(\mu_{FAC} - \mu_f\right)^2 \tag{2}$$

where the quadratic terms are the difference between FAC and vehicle position in coordinates of altitude, h, latitude, λ, and longitude, μ, respectively. This minimization problem is subject to kinematic and dynamic constraints and path constraints comprised of heat load, normal load, and dynamic pressure.

For a vehicle flying over a spherical, rotating Earth with an inverse-squared gravitational field, a 3-DOF model adequately represents its equations of motion (assuming zero sideslip angle). These equations are omitted for the sake of brevity and can be found in Ref. 26. The vehicle controls are the modulation of angle-of-attack, α, and bank angle, σ. For the complete flight path, the initial and final conditions are given by

$$(t_0, h_0, \mu_0, \lambda_0, V_0, \gamma_0, \xi_0) = (0 \text{ sec}, 167323 \text{ ft}, -85^\circ, 30^\circ, 8530 \text{ ft/s}, -1.3^\circ, 0^\circ)$$
$$(h_f, \mu_f, \lambda_f, V_f, \gamma_f, \xi_f) = (2000 \pm 400 \text{ ft}, -80.7112^\circ \pm 0.001371^\circ, 30^\circ \pm 0.001097^\circ, 300 \text{ ft/s}, -6.0^\circ, -60^\circ)$$

where V is the vehicle's velocity, γ is the flight-path-angle, and ξ is the heading angle. As seen in Fig. 5, the optimal trajectory obtained by using the above formulation satisfies all the constraints and successfully intercepts the center of the FAC. Figure 6 illustrates the intelligence of the system under the proposed method of trajectory generation. Here, various trajectories are generated from the same geographical location above the earth with initial velocities ranging from 400 ft/s to 2900 ft/s. As shown by the first trajectory (dark green) with an initial velocity of 400 ft/s, the trajectory emulates a typical "direct, straight-in" approach towards the runway. Having such limited energy, the vehicle guidance system (that hosts our optimal path planning method), intelligently chooses a path that flies directly to the runway in order to satisfy the required endpoint conditions. As the initial velocity (and thus, energy) increases, the trajectories evolve through more complex maneuvers as indicated by the path with the initial velocity of 1600 ft/s. In this case, the vehicle has much more energy than needed and the generated trajectory is an "S-turn"-like maneuver. For a vehicle with higher energy (e.g. initial velocity of 2800 ft/s), the planned path actually requires the vehicle to turn away from the runway and then loop back around as indicated by the last two trajectories. The blue circles in Fig. 6 represent the Heading Alignment Cylinders (HAC) used in Space Shuttle guidance.[27] The purpose of showing these is to see if there are any similarities between the optimal trajectories generated using the PS-based intelligent method and the typical trajectories that the Space Shuttle would track based on the HAC waypoints. Using a similar energy profile as of the Space Shuttle, Fig. 7 shows that the optimal trajectory generation produces trajectories that are in agreement with the HAC waypoints without an *a priori* knowledge of the HAC concept. This clearly demonstrates the power of using optimal control for autonomous and intelligent applications.

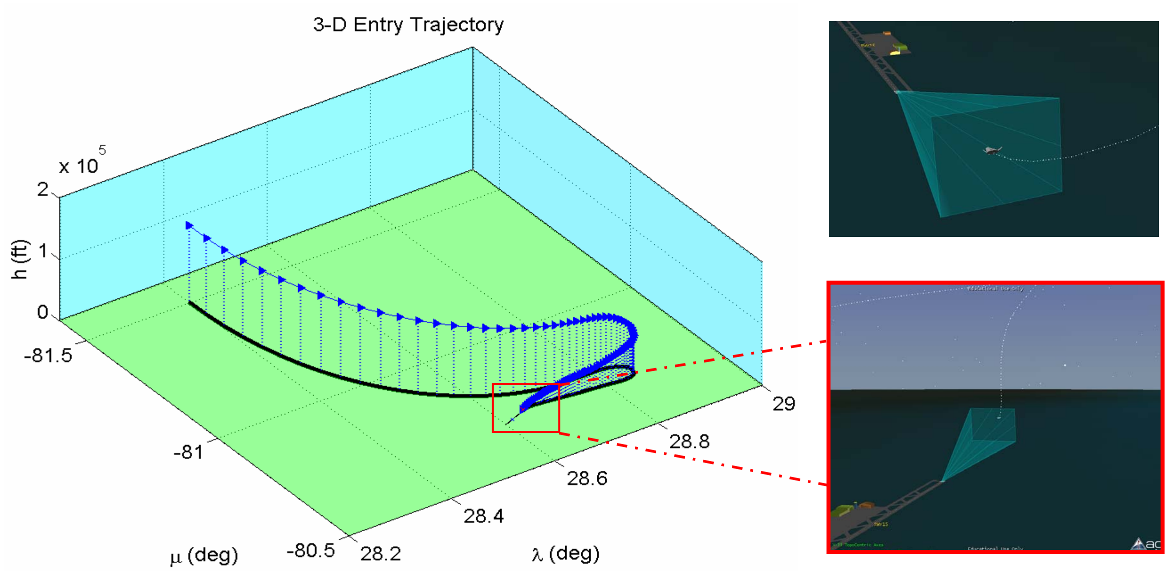


Figure 5. Reentry Trajectory Generated to Intercept Final Approach Corridor.

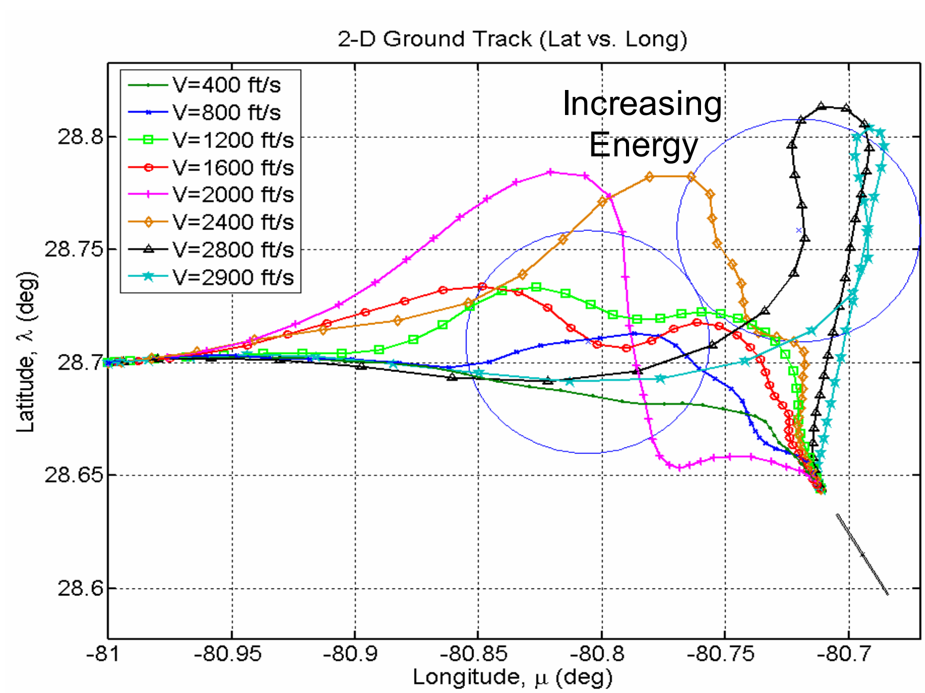


Figure 6. Intelligent Path Planning for Various Initial Velocities.

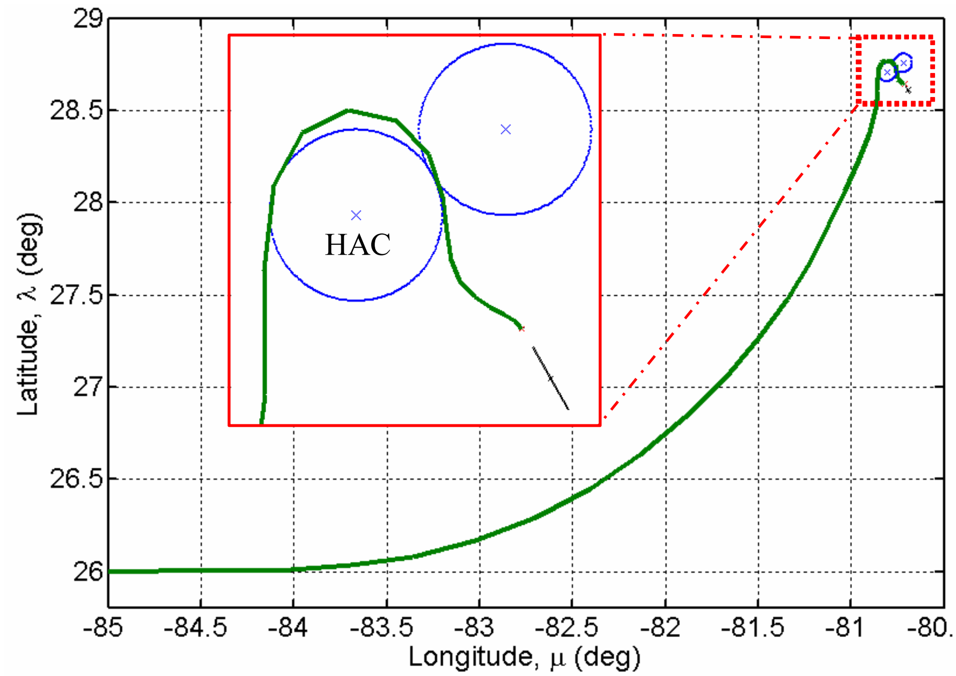


Figure 7. Intelligent Trajectory Generation and the Space Shuttle HAC.

One of the benefits of using the PS path planning method is that the LGL nodal points that are optimally selected by the algorithm[18] can be directly used as waypoints if it was desired to track the trajectories using other tracking control algorithms.

B. Ground Vehicle Path Planning in an Obstacle-Rich Environment

The next problem considered is the ground vehicle path planning in an obstacle-rich environment which is less complex, yet more familiar to the realm of autonomy. The nonholonomic kinematic equations are that of a four-wheeled car with constraints on the vehicle speed and steering wheels' angular deflection (front wheels only). Despite its apparent simplicity, many previous techniques have simplified the problem through linearization or avoidance of nonholonomic constraints.

Obstacles along the vehicle's path are modeled algebraically and include shapes such as diamonds, circles, ellipses, squares, and rectangles. The optimal control problem is defined as planning a successful and feasible path between a starting and finishing location while minimizing time, distance, or energy. Scenarios with moving obstacles, various obstacle configurations, and moving targets can be easily addressed by the proposed method of problem definition.[28] The results presented below display trajectories that satisfy minimum- time conditions.

Figure 8 portrays the time-optimal trajectories of the vehicle in obstacle-rich environments. In the left plot, 32 circular obstacles are considered. The resulting optimal path is intuitive. The car follows the shape of the obstacle when in close proximity and travels in a straight path when between obstacles. The right-hand plot displays a situation where a more kinematically complex path is generated as the time optimal solution. The non-intuitive solution is to travel below the large middle obstacle and includes backward motion. When we define the backward motion as not permissible, the trajectory generation scheme intelligently produces a different trajectory (with a higher cost) where the vehicle would travel around the left side of the large middle obstacle.

Parallel parking is another challenging problem for both autonomous and human-driven vehicles. Fig. 9 shows the applicability of this method to problems of this complex manner. The problem is posed as a vehicle traveling forward and approaching a spot in which it would like to parallel park. In most cases, drivers back their vehicles into the parking spot. In order to verify the intelligence of the scheme, in this case the trajectory planner is asked to park the car in the minimum amount of time without specifying how to do it. Again, the counter-intuitive result is to drive the car into the parking spot while moving forward and then correct the position of the car while in the spot (see Fig. 9).

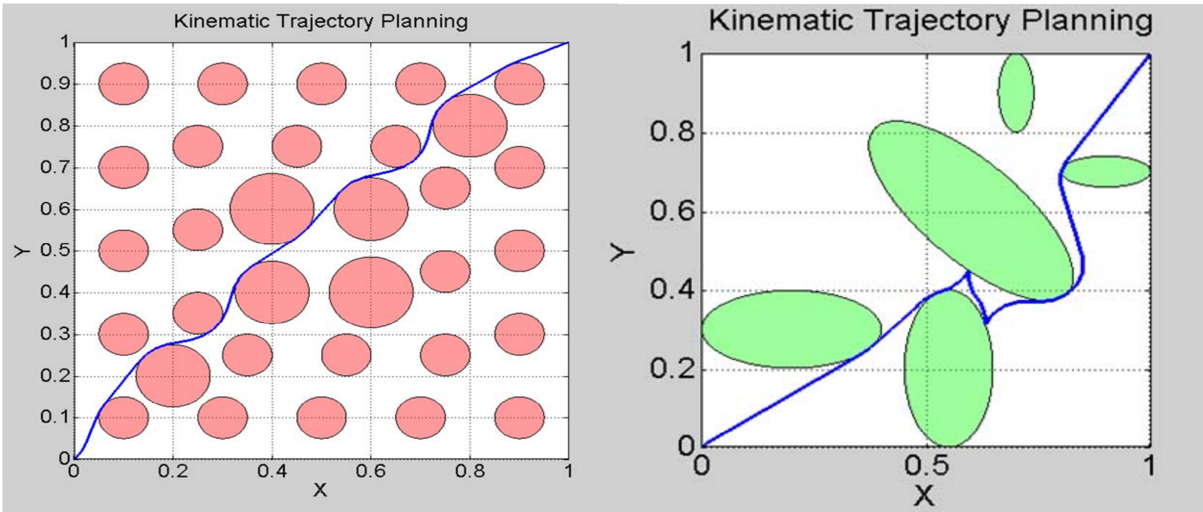


Figure 8. The Four-wheeled Car Executing a Minimum-time Maneuver in the Presence of Obstacles.

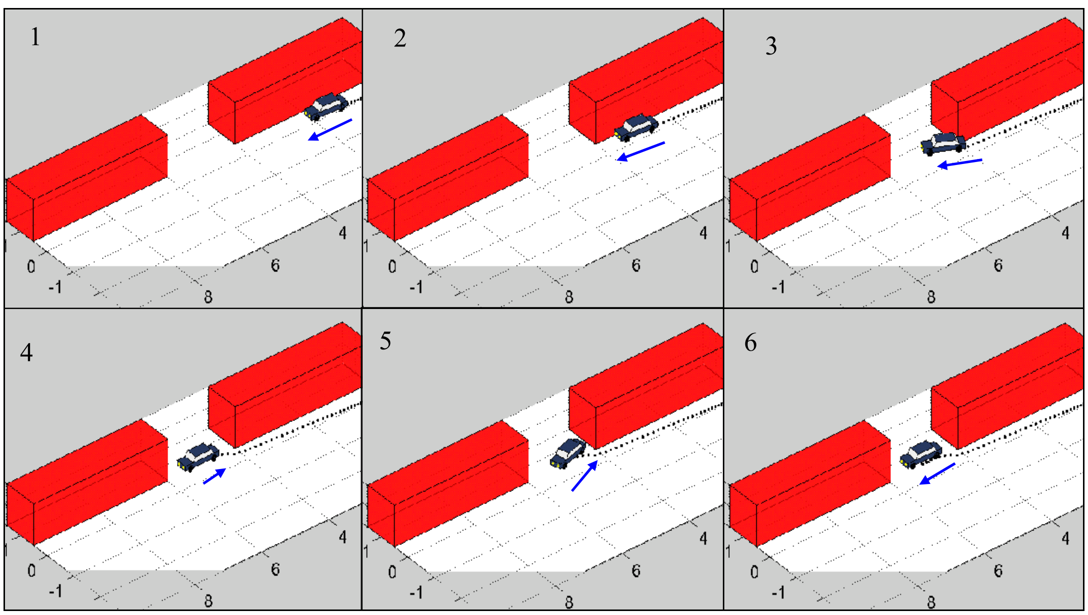


Figure 9. The Four-wheeled Car Executing a Parallel Parking Maneuver.

The problem framework can be adapted to three-dimensional UAV kinematics. Figure 10 shows a time-optimal kinematic trajectory of a UAV through an urban environment. This result could be uploaded and tracked by an onboard path follower. Future work will expand this research to real-time control trajectory generation. Instead of using the intelligent path planner to create the optimal path that must be tracked by an onboard controller, the optimal control trajectory will be re-generated in real-time during the UAV's flight. Finally, the real world demonstration of the method in practice will be accomplished using the AmigoBots shown in Fig. 4.

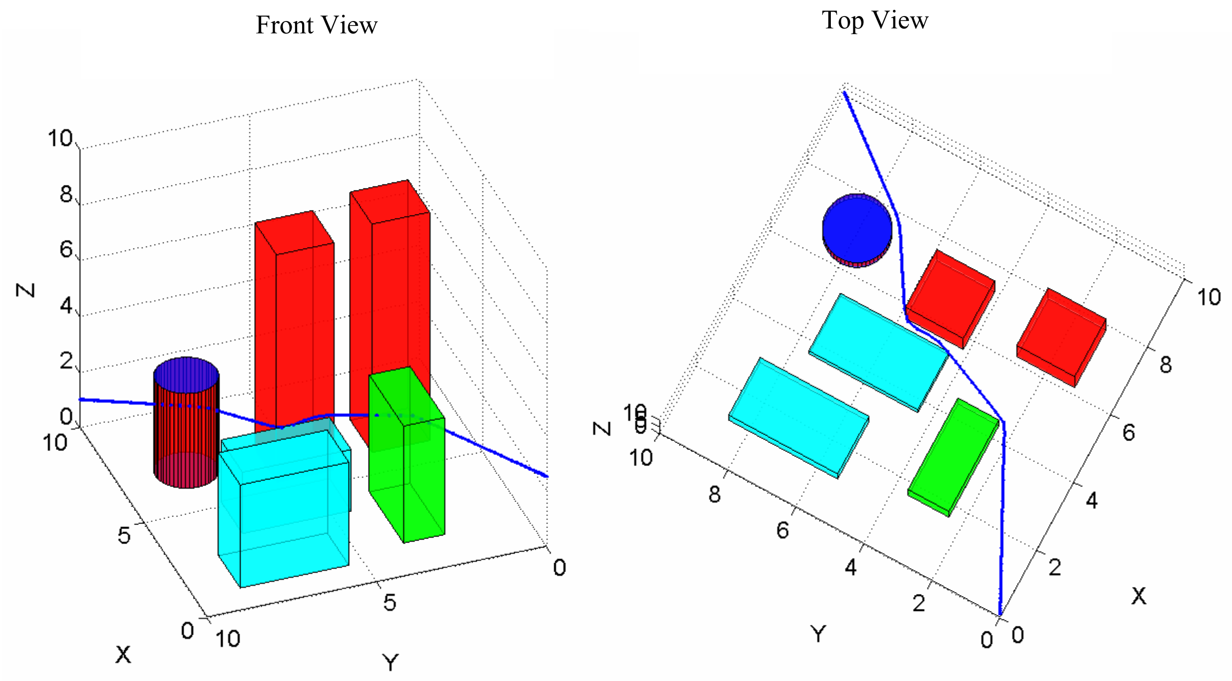


Figure 10. UAV Minimum-time Trajectory in an Urban Environment.

C. Loitering of Unmanned Aerial Vehicles

This example demonstrates the minimum-fuel path planning for high altitude circling flight of a UAV. Typically, fuel-optimal flights are designed for long range trajectories; however, UAV missions can also involve some form of circling flight in a prescribed area such as loitering over a target. Circling flight with constant radius is one such example. Since steady-state circling is known to be non-optimal[29], the question then becomes: what is the flight path that minimizes the fuel consumption. Using the variables shown in Fig. 11, the equations of motion for the UAV in circling flight can be represented by a simple point-mass drag polar aerodynamic model, as detailed in Ref. 30.

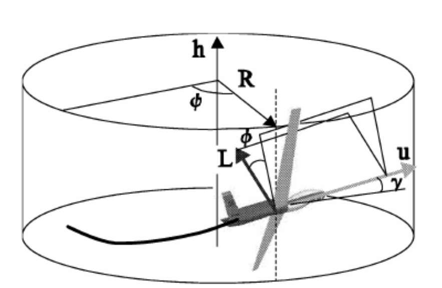
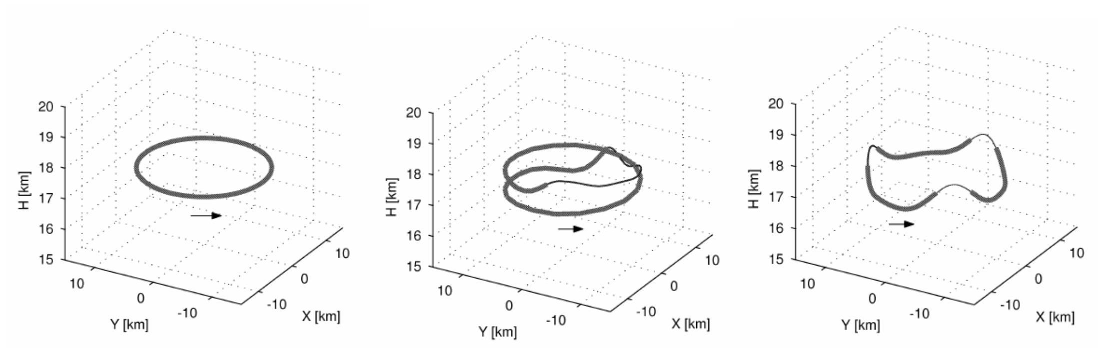


Figure 11. The Coordinate System and Reference Frame of a Circling UAV.

The controls are defined as thrust, lift coefficient, and bank angle, each constrained within minimum and maximum values based on the system's physical limitations. For example, the minimum thrust depends on the idle speed and the maximum thrust varies with altitude. In this example, the constraint set is varied based on the frequency of the periodic flight with a flight envelope limited to a cylinder with constant radius of 10 km. The optimal control problem is to minimize the fuel consumption

$$J = \frac{1}{t_f}\int_0^{t_f} cT(t)dt \tag{3}$$

where c*T represents the mass flow rate. This minimization is subject to the dynamic equations of motion, a constant radius cylinder, and a specified periodic frequency. The flight path results for various frequencies are shown in Fig. 12.



(a) (b) (c)

Figure 12. Flight Path Trajectories for Circling Flight at R=10 km: (a) Steady-State Circling (b) Periodic Circling of ½ Hz, (c) Periodic Circling of 3 Hz.

As illustrated, regardless of the desired periodic frequency and in the confines of a prescribed flight region, the path planning method intelligently determines the best way to fly in order to conserve fuel. With the optimal control trajectory provided, the on-board flight control system would only need to track this thrusting trajectory. This example can easily be expanded to investigating optimal flight paths in regions of varying shape and size.

D. Autonomous Spacecraft Slew Maneuver

Rapid spacecraft slew maneuvering is essential for spacecraft responsible for surveillance and reconnaissance. Despite previous works on showing that the time optimal maneuvers are not eigenaxis[31], researchers have been heretofore unable to solve the non-eigenaxis steering problem in a closed-loop format that would account for and counteract uncertainties and disturbances. This is, in part, due to the fact that classical closed-loop control techniques are, in general, ill suited for optimal slew maneuvering of systems with such nonlinearity. In this section, we show that the approach explained in section III provides a feedback control solution capable of autonomously counteracting exogenous disturbances and model uncertainties. This revolution in feedback control is obtained by recognizing that closed-loop solution does not necessarily imply closed-form solution. Provided open-loop optimal solutions could be obtained fast enough, successive re-computation of open-loop optimal solution will, in deed, provide on-board autonomous intelligence capability. Here, we demonstrate this concept by solving the closed-loop time-optimal slew problem for NPSAT1, an experimental spacecraft conceived, designed, and built at the Naval Postgraduate School, and scheduled to be launched in 2007.

The optimal control problem is to find the optimal control trajectory that minimizes the maneuver time for the spacecraft rest-to-rest reorientation maneuver given as

$$J = \int_{t_0}^{t_f} dt = t_f - t_0 \tag{4}$$

The kinematical and dynamical equations of motion are detailed in Ref. 32 and are not repeated here for the sake of brevity. The NPSAT1's three-actuation torques are generated as a result of interaction between the Earth's magnetic field and three onboard, magnetic torque-rods that can provide up to 30 $A \cdot m^2$; thus, $|u_i| \le 30 A \cdot m^2$. For a ϕ degree rest-to-rest roll maneuver (x-axis), the initial and final conditions are:

Initial conditions:

$$q_1(t_0) = q_2(t_0) = q_3(t_0) = 0, \quad q_4(t_0) = 1$$
$$\omega_x(t_0) = 0, \quad \omega_y(t_0) = 0, \quad \omega_z(t_0) = 0$$

Final conditions:

$$q_1(t_f) = \sin(\phi/2), \; q_2(t_f) = q_3(t_f) = 0, \; q_4(t_f) = \cos(\phi/2)$$
$$\omega_x(t_f) = \omega_y(t_f) = \omega_z(t_f) = 0$$

where $\boldsymbol{q} = \begin{pmatrix} q_1 & q_2 & q_3 & q_4 \end{pmatrix}^T$ are the quaternions and ${}^O\boldsymbol{\omega}_B^B = (\omega_x, \omega_y, \omega_z)^T$ is the angular rate of the body frame with respect to the orbit frame, expressed in the body frame.

To illustrate the performance of our proposed approach, the 135° rest-to-rest roll (x-axis slew) maneuver response of NPSAT1 is compared in three different motion scenarios: open-loop control for the ideal system (dashed lines), open-loop control for the real system with disturbances/uncertainties (thick dotted lines), and closed-loop control for the real system (solid lines). For the 135° maneuver, the final desired endpoint conditions are

$q^T(t_f) = [0.924 \quad 0 \quad 0 \quad 0.383]^T$. The NPSAT1 spacecraft parameters are listed in Table 1.

Table 1. Data for NPSAT1

Altitude	~ 560 km
Inclination	~ 35.4 deg
I_1	~ 5 kg.m^2
I_2	~ 5.1 kg.m^2
I_3	~ 2 kg.m^2
I_{12}= I_{13}= I_{23}	~ 0 kg.m^2
Max. Dipole Moment	~ 30 Amp.m^2

The open-loop time-optimal solution and the response of the ideal system (no disturbance or uncertainty) to such control trajectory are shown in Figs. 13 (dotted-lines) and 14 (dashed lines), respectively. It is clear that the time-optimal control solution is a bang-bang solution that brings the satellite to the final desired states (attitude and rates) within 220 seconds. The variations in q_1 and q_2 and the non-zero angular rates ω_1 and ω_2 indicate that the maneuver is not an eigenaxis slew. This shows the intelligence of the system not to follow the more intuitive eigen-axis maneuver. The tolerances on the acceptable final values for states (1 degree) and angular rates (0.06 deg/s) are shown by the dashed lines around each state.

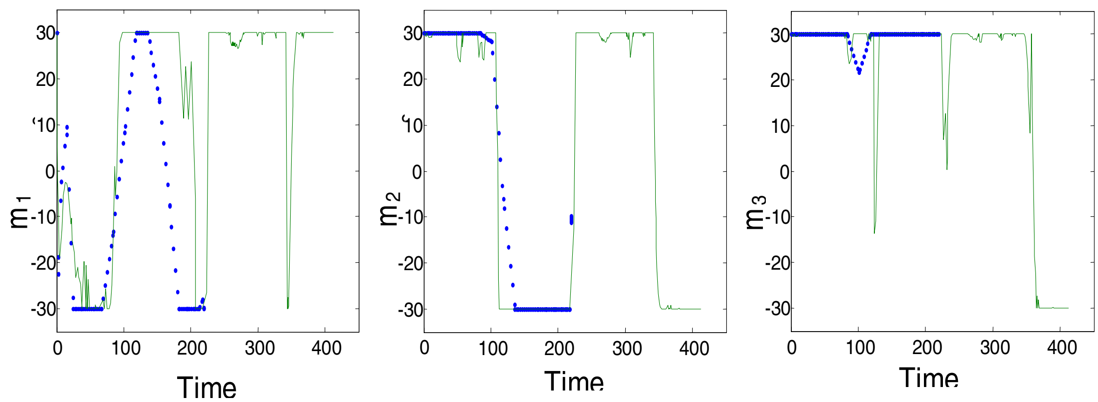


Figure 13. The Time-optimal Control Solution. *Dotted lines:* Open-loop; *Solid lines:* Closed-loop.

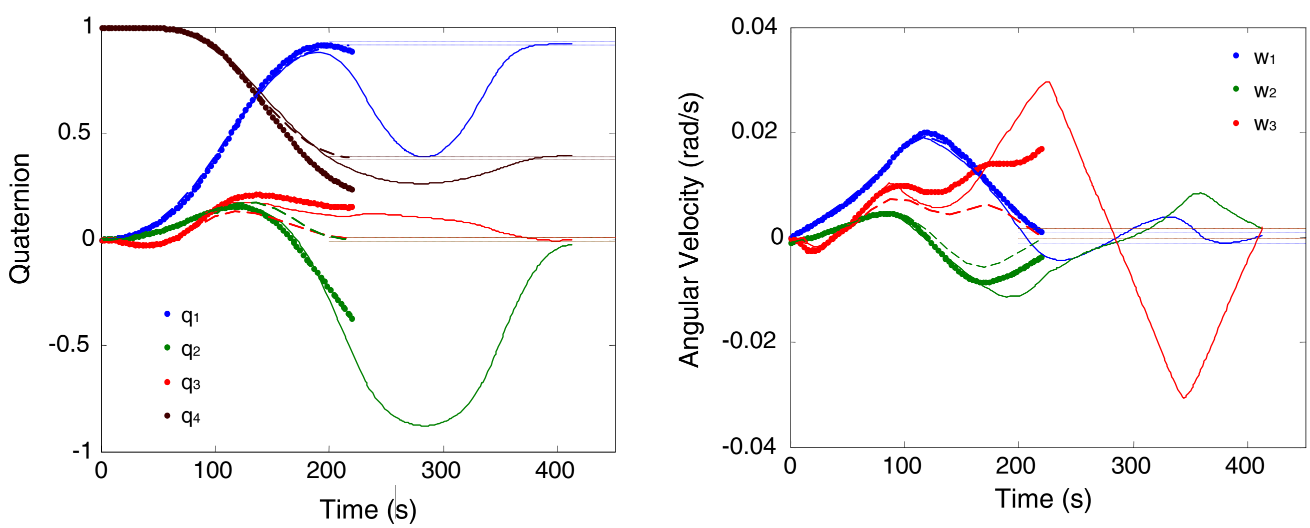


Figure 14. System Response. *Dashed:* Open-loop Control on Ideal System; *Thick dotted:* Open-loop Control on Real System; *Solid:* Closed-loop Control on Real System.

Next, we examine the performance of the open-loop solution in the presence of various disturbances. Disturbance in real applications is introduced as a result of parameter uncertainties, sensor measurement errors, or unknown external torques applied to the system. The dotted (thick) lines in Fig. 14 show the response of a real system to the same open-loop time-optimal control trajectory depicted in Fig. 13 (dashed lines). The moments of inertia of the real system are 5% off from the values used to generate the control and three external disturbance torques of $10.0 \text{x} 10^{-5} N.m$ are applied on the system for a period of 130 seconds (between t=50 s and t=180 s). Also, the numerical errors of state propagation play the role of sensor measurement errors in a real application. The dotted (thick) lines in Fig. 14 clearly confirm that application of the open-loop time-optimal solution to a real system fails to achieve the intended maneuver goals.

To counteract disturbance effects, our proposed intelligent control scheme is repeatedly recalculated and applied throughout the motion. Each new time-optimal trajectory is fed back to the system at each feedback instant. The overall closed-loop control seen in Fig. 13 (solid lines) is the amalgamation of the portions of such recalculated open-loop controls that are applied to the system autonomously. This results in a longer overall maneuver time (to counteract disturbances) and a completely different control trajectory as shown by the solid lines in Fig. 13. However, it provides additional intelligence to the control trajectory generation scheme that can now adjust its prior outputs (that were also optimal) based on the new circumstances. Solid lines in Fig. 14 show that the real (disturbed) system under such a uniquely computed control trajectory achieves the desired final states. Note that according to Bellman's Principle of Optimality, given an optimal open-loop trajectory, the re-optimized closed-loop trajectories should follow the original open-loop response. However, this is only valid for an ideal system without disturbance. Also, the PS approach to intelligent path planning and feedback control introduced in this paper does not require the prior knowledge of each "re-optimizing computation time" for state propagation. The current states of the system are obtained by propagating (under prior optimal control trajectory) prior known states. For the system under study, subsequent open-loop updates are computed within 0.84 to 10 seconds.

V. Conclusion

This paper demonstrated how PS-based optimal control can provide intelligence and autonomy for a broad range of unmanned applications. With such powerful methods readily available that provide real-time computational speed with current PC-computer technology, why settle for less than optimal? As implied by the various examples in this paper, optimality not only provides considerable savings in time, fuel, etc, but also improves safety, reliability, robustness, and autonomy as typically required of unmanned intelligent systems. In other words, optimal control theory is used not "merely" for optimality but also for autonomous path planning.

Acknowledgments

The authors would like to especially thank CDR Andrew Fleming (U.S. Navy) for initially addressing the open-loop optimal spacecraft slew maneuver and Dr. Harada Masanori (National Defense Academy, Japan) for his efforts on the UAV periodic circling problem. We also extend our appreciation to our numerous sponsors who supported the efforts in solving these types of problems throughout the years and welcome their continued interest and support of more challenging problems.

References

[1] Chien, S. and Hill, R. (NASA/ARC), Briefing on "Remote Agent", Mar 2000.

[2] DoD, "Joint Ground Robotics Enterprise," [online], http://www.jointrobotics.com, Oct 2006.

[3] Carroll, D.M., Nguyen, H.G., Laird, R.T., and Everett, H.R., "Center of Excellence for Small Robots," *Unmanned Ground Vehicle Technology VII, Proceedings of the SPIE*, Vol. 5804, pp. 501-510, May 2005.

[4] Cox, I.J. and Wilfong, G.T. (editors), *Autonomous Robot Vehicles*, Springer-Verlag, New York, NY, 1990.

[5] JAUS, "JAUS Tutorial," *Joint Architecture for Unmanned Systems*, [online], http://www.jauswg.org, Oct 2006.

[6] Ranganathan, A., and Kira, Z., "Self-Organization in Artificial Intelligence and the Brain," (unpublished), Georgia Tech [online], http://www.cc.gatech.edu/people/home/zkira/research.html, Apr 2007.

[7] Internet, http://www.webopedia.com, (last visited) Apr 2007.

[8] Dudek, G. and Jenkin, M., *Computational Principles of Mobile Robotics*, Cambridge University Press, UK, 2000.

[9] Barraquand, J., Langlois, B., and Latombe, J.C., "Numerical Potential Field Techniques for Robot Path Planning," *IEEE*, 1992.

[10] Vadakkepat, P., Tan, K.C., and Ming-Liang, W., "Evolutionary Artificial Potential Fields and Their Application in Real Time Robot Planning," *IEEE*, 2000.

[11] Wang, L.C., Yong, L.S., and Ang Jr., M.H., "Hybrid of Global Path Planning and Local Navigation Implemented on a Mobile Robot in Indoor Environment," *Proceedings of the 2002 IEEE International Symposium on Intelligent Control*, Vancouver, Canada, 27-30 Oct 2002.

[12] Zefran, M., *Continuous Methods for Motion Planning*, pp.1-14, University of Pennsylvania, Dec 1996.

[13] Choset, H. et al., *Principles of Robot Motion: Theory, Algorithms, and Implementations*, pp. 197-267 and 384-385, MIT Press, 2005.

[14] LaValle, S.M. and Kuffner, J.J., "Randomized Kinodynamic Planning," *International Journal of Robotics Research*," Vol. 20, No. 5, May 2001.

[15] Ross, I.M. and Fahroo, F., "A Unified Framework for Real-Time Optimal Control," *Proceedings of the IEEE Conference on Decision and Control*, Maui, Dec 2003.

[16] Sekhavat, P., Fleming A., and Ross, I. M., "Time-Optimal Nonlinear Feedback Control for the NPSAT1 Spacecraft," *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, AIM 2005, Monterey, CA, Jul 2005,.

[17] Gong, Q., Kang, W. and Ross, I. M., "A Pseudospectral Method for the Optimal Control of Constrained Feedback Linearizable Systems," *IEEE Transactions on Automatic Control*, Vol. 51, No. 7, pp. 1115-1129, Jul 2006.

[18] Ross, I.M., Sekhavat, P., Gong, Q., and Fleming, A., "Pseudospectral Feedback Control: Foundations, Examples and Experimental Results," *Proceedings of the 2006 AIAA Guidance, Navigation, and Control Conference*, AIAA Paper No. 2004-6354, Aug 2006.

[19] Bollino, K.P., and Ross, I.M., "A pseudospectral feedback method for real-time optimal guidance of reentry vehicles," *Proceedings of the 2007 IEEE American Control Conference*, New York, NY, Jul 2007 (to appear).

[20] Bryson, A.E., Ho, Y., *Applied Optimal Control: Optimization, Estimation, and Control,* Rev. Ed., Hemisphere Pub., New York, NY, 1975.

[21] Bryson, A.E., *Dynamic Optimization*, Addison Wesley Longman, Menlo Park, CA, 1999.

[22] Kirk, D.E., *Optimal Control Theory: An Introduction*, Prentice-Hall, Englewood Cliffs, 1970.

[23] Ross, I.M., *Control and Optimization: An Introduction to Principles and Applications, Electronic Edition*, Naval Postgraduate School, Monterey, CA, Dec 2005.

[24] Ross, I.M., *User's Manual for DIDO: A MATLAB Application Package for Solving Optimal Control Problems*, Tomlab Optimization, Sweden, Feb 2004.

[25] Gill, P.E., Murray, W., and Saunders, M.A., *User's Guide for SNOPT 5.3: A FORTRAN Package for Large-scale Nonlinear Programming*, Technical Report, Stanford Business Software, Inc., Palo Alto, CA, Jul 1998.

[26] Bollino, K., Ross, I.M. and Doman, D., "Optimal Nonlinear Feedback Guidance for Reentry Vehicles," *AIAA Guidance, Navigation and Control Conference*, AIAA Paper No. 2006-6074, Keystone, CO, 2006.

[27] NASA, *National Space Transportation System – Overview*, NASA History Division [online], http://history.nasa.gov/shuttleoverview1988/overview88.html, Sep 1988.

[28] Lewis, L.R., *Rapid Motion Planning and Autonomous Obstacle Avoidance for Unmanned Vehicles*, Master's Thesis, Naval Postgraduate School, Monterey, CA, Dec 2006.

[29] Ueno, S., "Minimum Fuel Cruise of Aircraft by Periodic Control," Journal of SICE, Vol.28, No.5, pp.604-609, 1992 (in Japanese).

[30] Harada, M., Bollino, K., and Ross, I.M., "Fuel Optimality of High Altitude Circling for Unmanned Aerial Vehicle," *2005 JSASS-KSAS Joint International Symposium on Aerospace Engineering*, Nagoya, Japan, Oct 2005.

[31] Bilimoria, K. D., and Wie, B., "Time-Optimal Three-Axis Reorientation of a Rigid Spacecraft," *Journal of Guidance, Control, and Dynamics,* Vol.16, No.3, pp. 446-452, 1993.

[32] Fleming, A., *Real-time Optimal Slew Maneuver Design and Control*, Astronautical Engineer's Thesis, US Naval Postgraduate School, 2004.