2015

Path planning and energy management of solar-powered unmanned ground vehicles

Adam Kaplan
Iowa State University

Path planning and energy management of solar-powered unmanned ground vehicles

by

Adam Kaplan

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Aerospace Engineering

Program of Study Committee:

Ran Dai, Major Professor

Christina Bloebaum

Ping Lu

Iowa State University

Ames, Iowa

2015

DEDICATION

This thesis is dedicated to my family and friends,
whose kindness, patience, support, and inspiration
have enabled this work.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisor, Dr. Ran Dai, for her support and tutelage over the past several years and for the trust she placed in me. I would also like to thank my committee members, Dr. Christina Bloebaum and Dr. Ping Lu, for their time and effort. Furthermore, I would like to thank staff of Automation and Optimization Laboratory - both past and present - as well as others who have helped make this work a reality, including Jeff Ellis, Nathaniel Kingry, Kishan Patel, Paul Uhing, and Justin Van Den Top.

ABSTRACT

Many of the applications pertinent to unmanned vehicles, such as environmental research and analysis, communications, and information-surveillance and reconnaissance (ISR), benefit from prolonged vehicle operation time. Conventional efforts to increase the operational time of electric-powered unmanned vehicles have traditionally focused on the design of energy-efficient components and the identification of energy efficient search patterns, while little attention has been paid to the vehicle's mission-level path plan and power management. This manuscript explores the formulation and generation of integrated motion-plans and power-schedules for solar-panel equipped mobile robots operating under strict energy constraints, which cannot be effectively addressed through conventional obstacle avoidance motion planning algorithms. Problem formulations are provided for transit problems conducted under both Balkcom-Mason and Pseudo-Dubins curves, as well as the generation of mission plans for vehicles which must persistently travel between certain locations. A comparison between one of the common motion-planning algorithms and experimental results of the prescribed algorithms, made possible by use of a test environment and mobile robot designed and developed specifically for this research, are presented and discussed.

CHAPTER 1. INTRODUCTION

Unmanned vehicles have demonstrated unique aptitude in multiple environments, and in a variety of applications. Unmanned Underwater Vehicles (UUVs) have been used for exploration, oceanography, underwater construction, and various military applications for a number of years (Robert W. Button (2009)). New Unmanned Ground Vehicles (UGVs) are currently being investigated by the United States military for cargo hauling (Marc Raibert and the BigDog Team (2008)), scouting, explosive detection and disposal, and other auxiliary warfighting tasks (Voth (2004)). Unmanned Aerial Vehicles (UAVs), well established as ideal platforms for remote sensing, mapping, and other tasks both civil and military, are now being applied to novel applications such as data-ferrying (Anthony J. Carfang and Kingston (2014)).

These types of missions can be broadly catagorized into three distinct groups - dull, dirty, and dangerous (Takayama et al. (2008); Gupta et al. (2013)). Dull missions include those which are long-duration and repetitive, such as information-surveillance-reconnaissance (ISR) and precision agriculture. Dirty missions are those which take place in unsafe or hazardous environments, including environmental research inside storms and hurricanes, as well as radiological events such as the Fukushima nuclear disaster. Dangerous missions require an unmanned vehicle to directly contend with other adversarial entities, and applies to a multitude of military applications such as suppression of enemy air defense (SEAD) missions.

Because dull missions, in particular, are characterized by prolonged periods of activity, they stand to greatly benefit from increased vehicular operation (endurance) and/or persistent operation. An unmanned vehicle's endurance, regardless of its environment, is inversely proportional to the consumption of its energy source (fuel, electrical power, etc.). Accordingly, an unmanned vehicle's endurance may be extended in two different ways - the vehicle could be used more efficiently, and/or the vehicle itself could be made more efficient.

Attempts to increase the endurance of unmanned vehicles have attracted attention from multiple academic circles, including the robotics community, and have in large part focused on the development and selection of efficient components (Vaussard et al. (2013)). Efficient path planning and search patterns in mobile robots, or UGVs, have also been investigated with some success (Wang et al. (2008); Mei et al. (2004)). Additionally, recent research into mission planning for multiple unmanned vehicles that work cooperatively to maintain a persistent mission presence has shown promise. For example, several authors have proposed using teams of unmanned vehicles which can adapt to new conditions or unanticipated failures in one of the member vehicles (Nigam (2014); Brett Bethke and Vian (2008); Mario Valenti and de Farias (2007)). Another work has investigated a swap and charge method for quickly replacing and recharging the batteries of a team of miniaturized unmanned aerial vehicles (UAVs) (N. Kemal Ure (2015)).

Although teams of unmanned vehicles may be feasible for some applications, they are still limited by the capacity of each vehicle to perform their tasks and move intelligently. Efforts to facilitate efficient and intelligent movement in unmanned vehicles have resulted in extensive work on motion planning algorithms, including A* (Zeng and Church (2009)), Probabilistic Road Maps (PRM) (Kavraki et al. (1996)), and Rapidly-exploring Random Trees (RRT) (Sertac Karaman (2011); Lawrance and Sukkarieh (2011)). These algorithms are commonly used in obstacle-avoidance problems to identify a vehicle's most efficient route. However, these methods are used are exclusively to determine the path a vehicle should take, and not the speeds at which the path should be traversed.

An alternative method to prolong a given vehicle's endurance is to harvest ambient environmental resources, such as solar energy, to act as a supplemental power supply. Recent work in developing long-duration solar-powered UGVs (Heng et al. (2011); Sulaiman et al. (2013); Lever et al. (2006)) which utilize the energy from the environment and charge storage batteries as backups for sustained operation has shown promise. For example, the Mars Opportunity Rover, which persistently explores unknown areas on Mars, has been working for a number of years using only solar power. Another example is the "cool robot" designed to use solar energy carry scientific payloads during summer in the Antarctic (Ray et al. (2007)).

While cloudless days and environments such as Mars or the Antarctic may be considered to have a uniform solar energy distribution for some period of time, many environments feature complex nonuniform and time-variant solar energy distributions, or insolations. Although dynamic power management microcontroller have been investigated and implemented in unmanned vehicles de J Mateo Sanguino and Gonzalez Ramos (2013), these controllers simply allocate power according to the solar energy available to the system at each moment, where available power is first allocated to the motors to maintain motion, and the remainder is used to charge a backup battery. The mere inclusion of a solar panel, then, is often insufficient to effectively extend a the endurance of an unmanned vehicle. Instead, the vehicle should be operated with respect to both the environment's insolation and the vehicle's overall mission objectives. This requires not only a deep understanding not only of the vehicle's power-consumption characteristics, but also of the environment's insolation.

Motion planning with respect to solar energy is a relatively new field. For instance, a mission planning algorithm for aircraft in persistent surveillance missions which gather solar energy along their designed paths has been demonstrated (Vasisht and Mesbahi (2015)), but assumes a constant insolation. This drawback is not particularly surprising, as characterizing an environment's insolation is in itself a great challenge - many environments feature nonuniform distributions which can not easily be expressed through fitted continuous functions, which precludes the use of gradient-based optimization techniques. Insolation models are often predictive, and are commonly developed using various forms of ray tracing (Glassner (1989)). Commercial software can generate digital elevation maps from aerial photographs, which can be used to construct insolation maps through ray tracing. Another method to develop insolation maps involves exploiting knowledge of the sun's current position and using an unmanned vehicle to identify an environment's shaded regions to calculate the height of obstacles within the region (Plonski et al. (2013); Plonski and Isler (2014)), which may also be used to create an insolation map via ray tracing.

This work contained in this manuscript describes mission planning techniques for an unmanned ground vehicle, equipped with a solar panel, which can be used to minimize the time required to accomplish a given movement, subject to various energy constraints, by optimiz-

ing both the vehicle's path and its velocities with respect to a model of the environment's insolation. These techniques employ modified variants of Particle Swarm Optimization (PSO) (Eberhart and Kennedy (1995)), a type of metaheuristic optimization, to address the difficulty of insolation characterization through continuous functions, and accounts for both the vehicle's performance and the availability of ambient solar energy in the vehicle's operating environment. The mission-planning techniques in this manuscript produce both path-plans and power-schedules. In conjunction, these schedules characterize the vehicle's path and the speeds at which it must travel and turn, unlike conventional motion planning approaches, which are used to produce a path free of environmental obstacles and do not determine the speed the vehicle travels at or the power it expends.

The mission planning techniques described in this manuscript are applied to two separate energy-constrained problems. The first of these problems is refereed to as the Transit Problem, wherein the unmanned vehicle must proceed from a given initial point to a desired ending location as quickly as possible while satisfying its energy constraint. This problem is formulated and solved with two types of curves. The first of these curves is the Balkcom-Mason (Balkcom and Mason (2002)), which is characterized by straight line-segments and zero-radius turns. The second curve is a modified form of the classical Dubins curve (Dubins (1957)), known as a Pseudo-Dubins curve. While the conventional Dubins curve is characterized by straight line movements and minimum radius circular-arc turns, the Pseudo-Dubins curve substitutes the traditional uniform minimum turning radii with individually designed turning radii for each turn. For each of these curves, a similarly-structured optimization problem is posed and solved with a modified PSO.

The second of these problems is the Persistent Traveling Vehicle problem, where the vehicle must visit a number of locations and return to its starting position and orientation, subject to the same energy constraints as earlier. In many ways this resembles the classical Traveling Salesperson Problem (TSP), which requires a salesperson to determine what order to visit some number of cities so as to minimize the total distance traveled. Solving the TSP is computationally challenging, and has spurred the development of multiple solution algorithms. Because the Persistent Traveling Vehicle problem requires both the determination of what order to visit cer-

tain locations in the vehicle's environment, as well as the velocities at which to travel and turn, it is even more computationally difficult. This problem is solved under the Balkcom-Mason regime through a cascaded optimization process which employs both variants of the modified Particle Swarm Optimizer used to solve the Transit Problem as well as typical optimization techniques for the Traveling Salesperson Problem.

The organization of this thesis is as follows: Chapter 2 contains problem formulations for the energy-constrained Transit Problem under both the Balkcom-Mason and Pseudo-Dubins curves, including the vehicle's dynamics, energy harvesting, and power consumption models. Chapter 2 also introduces the modified Particle Swarm Optimization process used to solve the Transit Problem. Chapter 3 provides the problem formulation for the energy-constrained Persistent Traveling Vehicle, and explores the Cascaded Particle Swarm Optimization process and its application to the problem. Chapter 4 contains a discussion of the hardware and test environment developed to evaluate these methods, as well as a method of characterizing the test environment's insolation. Chapter 5 contains both simulation and experimental results of the methods discussed in Chapters 2 and 3. Finally, this manuscript's conclusions and closing remarks are presented in Chapter 6.

CHAPTER 2. ENERGY-CONSTRAINED TRANSIT PROBLEM

In this Chapter, we address the energy-constrained Transit Problem, in which an unmanned ground vehicle, equipped with a solar-panel, is required to traverse an area of known insolation as quickly as possible, constrained so that the vehicle's net energy gain at the end of its prescribed movements is greater than or equal to zero. It is assumed that the vehicle's performance characteristics have been previously established, and also that the environment is of level, uniform terrain. Two types of curves, Balkcom-Mason and Pseudo-Dubins, are first discussed in sections 2.1 & 2.2 (Kaplan et al. (2015b,a)). An optimization problem which can be applied to either of these curves is then formulated in Section 2.3, along with the vehicle's energy harvesting, and power consumption models. Section 2.4 proceeds to offer a brief introduction to Particle Swarm Optimization, and Section 2.5 concludes with a modified Particle Swarm Optimization algorithm which may be used to solve the optimized problem posed earlier in the Chapter.

2.1 Balkcom-Mason Curve Formulation

We begin by examining a Balkcom-Mason type curve (Balkcom and Mason (2002)), which is characterized by straight-line segments and zero-radius turns. Balkcom-Mason curves permit the vehicle to travel forward and backwards in straight lines, and turn in-place both clockwise and counterclockwise. In this particular problem, we may safely restrict the vehicle to forward straight-line movement and zero-radius turns in both directions, as demonstrated on the following page in Fig. 2.1.

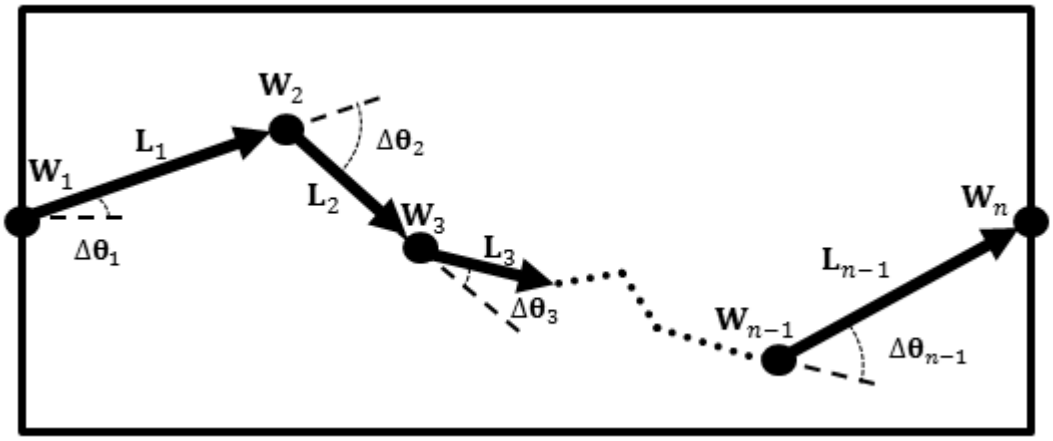

Figure 2.1 Path segments of the Balkcom-Mason curve.

2.2 Pseudo-Dubin Curve Formulation

We now proceed to examine a Pseudo-Dubins curve, which substitutes the traditional uniform minimum turning radii of the classical Dubins curve with individually designed turning radii. In this case, the UGV's movement is governed by three differential-drive primitives including straight lines, where both drive wheels move the same speed and direction, and circular-arc turns in both the clockwise and counterclockwise directions, where both drive wheels move in the same direction but at different speeds. While a classical Dubins curve is characterized by circular turns using the vehicle's minimum turning radius, imposing such a constraint on a vehicle capable of performing zero radius turns yields a Balkcom-Mason curve, discussed in the Section above. Allowing the radii of the UGV's turns to vary not only allows the principles of the Dubins curve to be applied to a differential drive vehicle, but also allows the UGV to continue progressing towards its destination while changing its orientation. In the experimental results presented in Chapter 5, the Pseudo-Dubins curve requires less power than the zero-radius turns employed under the Balkcom-Mason curve.

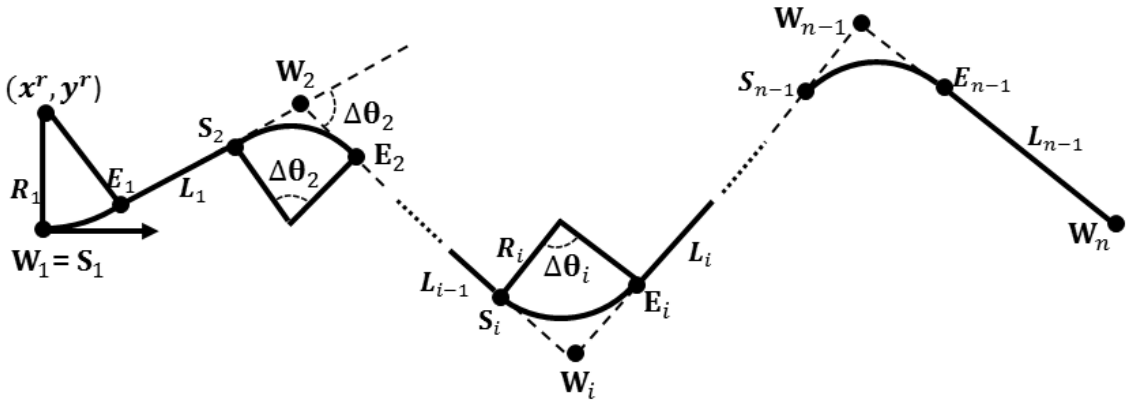

Figure 2.2 Path segments of the Pseudo-Dubins curve.

2.3 Optimization Problem Formulation

For both the Balkcom-Mason and Pseudo-Dubins curves, the path from the initial point W_1 to the final point W_n is composed of $n-2$ interception points and $n-1$ turn and line pairs. To simplify the path planning problem, the design variables for both curves are comprised of the intersection points, denoted as $W_i(x_i, y_i)$, $i = 2, \ldots, n-1$, the linear speed along the line segment, denoted as V_i, $i = 1, \ldots, n-1$, and the angular speeds along the turn or turn segment, denoted as ω_i, $i = 1, \ldots, n-1$.

Determining the beginning and ending locations of each turn, as well as the change in orientation $\Delta\theta(i)$, for a Balkcom-Mason curve is trivial. For the Pseudo-Dubins curve, each starting point S_i and ending point E_i of circular-arc i for $i = 2, \ldots, n-1$ are determined by

$$x_{S_i} = x_i - R_i \cos\theta_i,\ y_{S_i} = y_i - R_i \sin\theta_i, \tag{2.1}$$

$$x_{E_i} = x_i + R_i \cos\theta_i,\ y_{E_i} = y_i + R_i \sin\theta_i, \tag{2.2}$$

$$\forall\, i = 2, \ldots, n-1,$$

where R_i is the turn radius of circular-arc i, and is predefined by a polynomial function with respect to the angular velocity, denoted as $f_R(\omega_i)$. The heading angle θ_i along line segment i for $i = 2, \ldots, n-1$ is found from

$$\theta_i = \tan^{-1} \frac{y_{i+1} - y_i}{x_{i+1} - x_i},\ \forall\, i = 2, \ldots, n-1. \tag{2.3}$$

For the first circular-arc, $i = 1$, S_1 coincides with the initial point W_1, such that $x_{S_1} = x_1$ and $y_{S_1} = y_1$. Subsequently, the coordinate of E_1 are determined by

$$x_{E_1} = x_2 - sign(x_2 - x_1)d_1 \cos(\tan^{-1}\frac{y_2 - y^r}{x_2 - x^r} - \sin^{-1}\frac{R_1}{d_1}), \tag{2.4}$$

$$y_{E_1} = y_2 - sign(y_2 - y_1)d_1 \sin(\tan^{-1}\frac{y_2 - y^r}{x_2 - x^r} - \sin^{-1}\frac{R_1}{d_1}), \tag{2.5}$$

where $x^r = x_1$ and $y^r = y_1 + sign(y_2 - y_1)R_1$ are the coordinates for the center of the first circular-arc, R_1 can again be determined from the polynomial function $f_R(\omega_1)$, and $d_1 = \sqrt{(y_2 - y^r)^2 + (x_2 - x^r)^2}$.

For both curves, the time required for the UGV to complete the the prescribed turns and line segments is expressed as

$$T_{total} = \sum_{i=1}^{n-1}(T^r(i) + T^l(i)), \tag{2.6}$$

where the superscripts 'r' and 'l' denote variables associated with the turn and line segments, respectively. With the corresponding rotational and linear velocities, ω_i and V_i, the total travel time can be reformulated as

$$T_{total} = \sum_{i=1}^{n-1}(\frac{\Delta\theta_i}{\omega_i} + \frac{L_i}{V_i}), \tag{2.7}$$

where $\Delta\theta_i = \theta_{i+1} - \theta_i$, $i = 1, \ldots, n-1$, is the change in heading angle along turn i, while the length of line-segment i, L_i, is simply $L_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$ for the Balkcom-Mason curve and $L_i = \sqrt{(x_{S_{i+1}} - x_{E_i})^2 + (y_{S_{i+1}} - y_{E_i})^2}$ in a Pseudo-Dubins curve.

In both curves, the energy gathered along each turn and line segment is an integral function of the energy collection rate, P_{in}, over the relative time span, where P_{in} is dependent on the solar radiation strength over the concerned area, $R_{in}(a, b)$, and the solar panel area of the UGV, denoted as A_s.

Recall that the environment's solar energy density can not be expediently represented as a continuous function. As a result, the harvested energy can not be integrated analytically. Instead, we place Q equidistantly spaced samples across R_{in} from the starting to the ending points along each turn and line segment i to evaluate the average solar radiation strength along the line segment, and multiply it by the time spent by the vehicle performing the movement.

The energy gathered from each line and turn segment may me formulated as

$$E^l_{in}(i) = \sum_{q=1}^{Q} R_{in}(a_q, b_q) A_s L_i/(QV_i) \tag{2.8}$$

$$E^r_{in}(i) = \sum_{q=1}^{Q} R_{in}(a_q, b_q) A_s \Delta\theta_i/(Q\omega_i). \tag{2.9}$$

The energy consumed by the UGV to complete each movement is determined by the engines' power consumption rate $P^l_e(i)$ and $P^r_e(i)$, $e = 1, \ldots, N_e$, where N_e is the number of engines, as well as the power required by the on-board components (microcontroller, wireless modem, voltage/current sensors, etc.), and the time required to finish the corresponding segment. Each engine's consumption rate under both linear and rotational movements is a predefined polynomial function of corresponding linear and angular speed, denoted as $P^l_e(V_i)$ and $P^r_e(\omega_i)$, $e = 1, \ldots, N_e$, respectively. In order to simplify the expression, we assume the power controller will always allocate the necessary power to the microcontroller, wireless sensor, and other miscellaneous components. The summation of the three consumption units is denoted as the vehicle's passive power consumption, P_a. Mathematically, the consumed energy is represented by

$$E^l_{out}(i) = (\textstyle\sum_{e=1}^{N_e} P^l_e(V_i) + P_a) T^l(i) \tag{2.10}$$

$$E^r_{out}(i) = (\textstyle\sum_{e=1}^{N_e} P^r_e(\omega_i) + P_a) T^r(i) \tag{2.11}$$

To satisfy the energy constraint at the final point, W_n, such that the net energy change ΔE_{total} at the end of the vehicle's movement is above zero, we have

$$\Delta E_{total} \quad = \quad \sum_{i=1}^{n-1} (E^l_{in}(i) + E^r_{in}(i) - E^l_{out}(i) - E^r_{out}(i)) \geq 0. \tag{2.12}$$

The engines' power may be supplied from two sources - the harvested energy from environment, and/or the backup battery. At each movement segment, the power controller allocates the necessary power to each engine to achieve the vehicle's desired motion. Therefore, the power among all the electric components is balanced by

$$\sum_{e=1}^{N_e} P^l_e(V_i) + P_a + P^l_b(i) = \sum_{q=1}^{Q} R_{in}(a_q, b_q) A_s/Q, \tag{2.13}$$

$$\sum_{e=1}^{N_e} P_e^r(\omega_i) + P_a + P_b^r(i) = \sum_{q=1}^{Q} R_{in}(a_q, b_q) A_s/Q, \tag{2.14}$$

where the battery consumed/supplied power, $P_b^l(i)$ and $P_b^r(i)$ is constrained by its capacity, $P_{b_{max}}$ and $P_{b_{min}}$, such that $P_{b_{min}} \leq P_b^l(i) \leq P_{b_{max}}$ and $P_{b_{min}} \leq P_b^r(i) \leq P_{b_{max}}$.

With the path, energy harvesting, and consumption models described above, the integrated path planning and power management optimization problem for both the Balkcom-Mason and Pseudo-Dubins curves can be formulated as

$$\begin{aligned} & \min_{\mathbf{X}} \textstyle\sum_{i=1}^{n-1} (T^l(i) + T^r(i)) \\ s.t. \quad & -\textstyle\sum_{i=1}^{n-1} E_{in}^l(i) + E_{in}^r(i) - E_{out}^l(i) - E_{out}^r(i) \leq 0 \\ & \textstyle\sum_{e=1}^{N_e} P_e^l(V_i) + P_a + P_b^l(i) - \sum_{q=1}^{Q} R_{in}(a_q, b_q) A_s/Q = 0 \\ & \textstyle\sum_{e=1}^{N_e} P_e^r(\omega_i) + P_a + P_b^r(i) - \sum_{q=1}^{Q} R_{in}(a_q, b_q) A_s/Q = 0 \\ & P_{b_{min}} \leq P_b^l(i) \leq P_{b_{max}} \\ & P_{b_{min}} \leq P_b^r(i) \leq P_{b_{max}} \end{aligned} \tag{2.15}$$

where the variable set to be optimized, $\mathbf{X}$, includes the coordinates of $n-2$ interception points (x_i, y_i), $i = 2, \ldots, n-1$, and the $n-1$ linear, V_i, and rotational, ω_i, speeds along each segment i, $i = 1, \ldots, n-1$.

2.4 Particle Swarm Optimization

Particle Swarm Optimization, first introduced by Eberhart and Kennedy in 1995, is a heuristic optimization method capable of exploring the design spaces of both continuous and non-continuous problems. Particle Swarm Optimization iteratively evaluates and adjusts a number k of possible solutions to an optimization problem, represented as particles. These "particles" exist in a space consisting of as many dimensions d as there are optimization variables, such that each particle represents a distinct solution to the optimization problem

$$P^j = [x_1^j, x_2^j, \ldots, x_d^j], \forall\, j = 1, \ldots, k. \tag{2.16}$$

These particles are iteratively perturbed a varying distance M in each dimension i in the direction of the particle's personal best performing location, P_{BEST}, and the group's single

best performing location, G_{BEST}, influenced by the local and global confidence factors C_1 and C_2. Both movements are multiplied by a random number R between 0 and 1. Each of the particle's subsequent movements in a given dimension are influenced by the previous movement $M_i^j(t-1)$, multiplied by an inertia factor I. Mathematically, M_i^j is expressed as

$$M_i^j = IM_i^j(t-1) + (C_1R(P_{BEST_i}^j - P_i^j)) + (C_2R(G_{BEST_i} - P_i^j)). \tag{2.17}$$

Each particle is reevaluated after its movement, and if appropriate, replaces its previous P_{BEST} location and the group's G_{BEST}. Subsequent updates to G_{BEST} will identify a progressively optimal solution set, eventually leading to a local or global extrema. Upon initialization, each particle's P_{BEST} must be its present position. As a result, the initial movement expression $M_{initial}$ may be simplified as

$$M_{i\ initial}^j = C_2R(G_{BEST_i} - P_i^j). \tag{2.18}$$

For the particle whose initial position was found to be the best initial global best, $P_i^j = G_{BEST_i}$, and subsequently, $M_{i\ initial}^j = 0$.

2.5 Modified Particle Swarm Optimization for Mission Planning

In the integrated path planning and power management problem formulated in (2.15), each particle P consists of $n-2$ interception points $W_i = (x_i, y_i)$, $i = 2, \ldots, n-1$, and $n-1$ pair of speed settings $U_i = (V_i, \omega_i)$, $i = 1, \ldots, n-1$, summarized as

$$\begin{aligned} P^j &= [W_2^j, \ldots, W_{n-1}^j, U_1^j, \ldots, U_{n-1}^j] \\ &= [x_2^j, y_2^j, \ldots, x_{n-1}^j, y_{n-1}^j, V_1^j, \omega_1^j, \ldots, V_{n-1}^j, \omega_{n-1}^j]. \end{aligned}$$

Because the dimension of each particle directly correlates to the number of intersection points n, a small n will decrease the PSO's computation time, but also may lead to an infeasible solution. Conversely, specifying a large n will increase the computation time in exchange for a greater likelihood of a feasible solution with an improved performance index. Therefore, the

dimension n must be carefully chosen based off of the physical dimensions of the problem area and the sparsity of the available solar energy.

In a conventional PSO, the P_{best} and G_{best} updates occur only when a particle's performance index is greater than the previous P_{best} or G_{best}, and when the particle meets all of the problem's constraints. In our problem, P_{best} and G_{best} updates would happen only when a particle's T_{total} is less than the previous P_{best} or G_{best} value and when the particle's ΔE_{total} satisfies the energy constraint, expressed as

$$P^j_{best} = P^j, \text{ if } [T_{total}(P^j_{best}) > T_{total}(P^j)] \,\&\, [\Delta E_{total}(P^j) > 0],$$
$$G_{best} = P^j, \text{ if } [T_{total}(G_{best}) > T_{total}(P^j)] \,\&\, [\Delta E_{total}(P^j) > 0].$$

Due to the strict constraints in the above update conditions, it is difficult to generate random particles in the initialization step to guarantee that each particle satisfies the energy constraints, and difficult to iterate the particles towards increasingly optimal solutions which satisfying the constraints. In this method, we impose a new P_{best} update relationship to promote gradual fulfillment of the problem's energy constraint. At the initialization step, each P_j is pseudo randomly populated such that $x_1 < x^j_2 < \ldots < x^j_{d-1} < x_d$ and all W and U values fall within permissible ranges originally defined by the problem. Each particle's location, P^j, and corresponding objective value, $T_{total}(P^j)$, are saved regardless of the values of $\Delta E_{total}(P^j)$. If any particle exists with a $\Delta E_{total} > 0$, G_{best} will be selected from that group by looking for the lowest T_{total}. If none of the initialized particles meet the energy constraint, then G_{best} will be selected as the particle with the ΔE_{total} closest to zero. Once they have been initialized, each particle is iteratively perturbed and evaluated against both its P_{best} and the group's G_{best}, and if necessary, they are replaced by

$$P^j_{best} = P^j, \text{ if } [T_{total}(P^j_{best}) > T_{total}(P^j) \,\&\, [\Delta E_{total}(P^j) > 0],$$

or

$$P^j_{best} = P^j, \text{ if } [\Delta E_{total}(P^j) > \Delta E_{total}(P^j_{best})] \,\&\, [\Delta E_{total}(P^j_{best}) < 0],$$

and

$$G_{best} = P^j, \text{ if } [T_{total}(G_{best}) > T_{total}(P^j_{best})] \,\&\, [\Delta E_{total}(P^j) > 0].$$

By imposing this dual P_{best} update relationship, we allow particles that are not necessarily power consistent upon initialization to gradually move towards satisfying the problem's energy constraint and help ensure that G_{best} values meet the problem's constraints. Once a particle has achieved energy consistence, the first update relationship continues to drive it towards a minimal T_{total}.

CHAPTER 3. ENERGY-CONSTRAINED PERSISTENT TRAVELING VEHICLE PROBLEM

This Chapter discussed the energy-constrained Persistent Traveling Vehicle Problem, which requires a solar-panel equipped ground vehicle to visit a series of objective points and return to its starting location and orientation as quickly as possible, constrained such that the vehicle's net energy gain must be greater than or equal to zero (Kaplan et al. (2016)). Recall that this problem is analogous, in a general sense, to the classical Traveling Salesperson Problem (TSP), but more complex in that we must solve for both the visiting sequence to each objective point as well as the motion and power allocation plans to optimize the vehicle's travel time subject to its net energy constraint. This complexity requires a novel approach to effectively plan the vehicle's motion and power schedule. The Persistent Traveling Vehicle optimization problem, along with a brief review of the vehicle's dynamics and energy models, are presented in Section 3.1, and a new cascaded, or sequential, optimization process is presented in Section 3.2.

3.1 Problem Formulation

Consider the same solar-panel equipped UGV from the previous Chapter, which must now travel to a number of objective points and then return to its starting position and orientation in the shortest amount of time. In order to persistently repeat this mission, the vehicle's net energy gain at the moment it returns to its starting orientation must be greater than or equal to zero. Let this vehicle's movement be governed by the four differential-drive primitives of a Balkcom-Mason curve. Unlike the scenario described in Section 2.1, there is no requirement that the vehicle must always travel forwards. It may be assumed that the vehicle may drive backwards at the same speed and for the same energy consumption as driving forwards. Sub-

sequently, the vehicle shall be permitted to travel forwards and backwards, as well as turn clockwise and counterclockwise. The UGV may travel directly between each objective point, denoted by $O_1(x_1, y_1), \ldots, O_m(x_m, y_m)$, where m is the number of objective points, or elect to detour to other locations to gather energy. The waypoints visited by the vehicle, denoted by $W_1(x_1, y_1), \ldots, W_n(x_n, y_n)$, where n is the number of waypoints, must include the set of objective points, $O \subseteq W$, as demonstrated in Figure 3.1. In this problem, the vehicle must

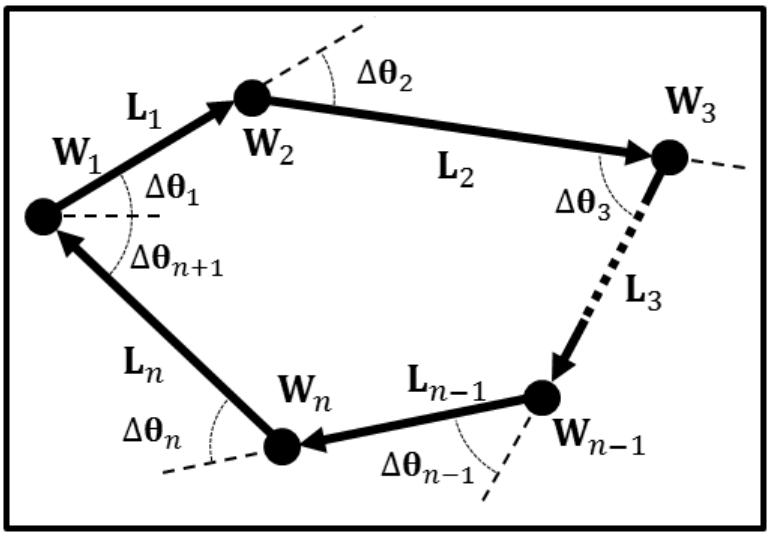

Figure 3.1 Example for motion of n waypoints.

travel from $W_1(x_1, y_1)$ to $W_n(x_n, y_n)$ and then return to $W_1(x_1, y_1)$. This requires the vehicle to traverse a total of n line segments and $n+1$ turns. Accordingly the time spent by the UGV traveling through the designed path be expressed as

$$T_{total} = \sum_{i=1}^{n+1}(T^r(i)) + \sum_{i=1}^{n}(T^l(i)). \tag{3.1}$$

Similarly to equation 2.7 from Section 2.3, T_{total} can be rewritten as

$$T_{total} = \sum_{i=1}^{n+1}(\frac{\Delta\theta(i)}{\omega(i)}) + \sum_{i=1}^{n}(\frac{L(i)}{V(i)}), \tag{3.2}$$

where $\Delta\theta(i)$ is the required change in angle from the vehicle's present heading to the new heading before moving straightly to the next waypoint, and $L(i)$ is the length of line-segment i. The vehicle's energy in, energy out, and power balance equations are identical to those presented in Section 2.1 (equations 2.8, 2.10, and 2.13, respectively). However, because the

vehicle must traverse n lines and $n+1$ turns, the ΔE_{total} equation must be reformatted as

$$\Delta E_{total} = \sum_{i=1}^{n+1}(E_{in}^{r}(i) - E_{out}^{r}(i)) + \sum_{i=1}^{n}(E_{in}^{l}(i) - E_{out}^{l}(i)) \geq 0 \tag{3.3}$$

The motion planning problem for the energy-constrained persistent traveling vehicle can now be summarized as an optimization problem in the form of

$$\min_{\mathbf{X}} \textstyle\sum_{i=1}^{n+1} T^{r}(i) + \sum_{i=1}^{n} T^{l}(i) \tag{3.4}$$

$$\text{s.t.} \quad O \subseteq W$$

$$W(1) - W(n) = 0 \tag{3.5}$$

$$-\textstyle\sum_{i=1}^{n+1}(E_{in}^{r}(i) - E_{out}^{r}(i)) - \sum_{i=1}^{n}(E_{in}^{l}(i) - E_{out}^{l}(i)) \leq 0 \tag{3.6}$$

$$\textstyle\sum_{e=1}^{N_e} P_e^{l}(V_i) + P_a + P_b^{l}(i) - \sum_{q=1}^{Q} R_{in}(a_q, b_q)A_s/Q = 0$$

$$\textstyle\sum_{e=1}^{N_e} P_e^{r}(\omega_i) + P_a) + P_b^{r}(i) - \sum_{q=1}^{Q} R_{in}(a_q, b_q)A_s/Q = 0$$

$$P_{b_{min}} \leq P_b^{l}(i) \leq P_{b_{max}}$$

$$P_{b_{min}} \leq P_b^{r}(i) \leq P_{b_{max}}$$

where the variable set to be optimized, $\mathbf{X}$, includes the coordinates of $n-2$ interception points $W(x_i, y_i)$, $i = 2, \ldots, n-1$, the n linear speeds V_i, $i = 1, \ldots, n$, and the $n+1$ angular speeds ω_i along each turn, $i = 1, \ldots, n+1$.

3.2 Cascaded Optimization for Mission Planning

The motion planning problem formulated above is a mixed-integer nonlinear optimization problem which includes both integer and continuous variables, and as a result, it is difficult to find an optimal solution within a reasonable amount of computation time. The motion planning problem requires that the waypoints visited by the UGV, $W_i(x_i, y_i)$, $i = 1, \ldots, n$, include all of the mission's objective points, $O_j(x_j, y_j)$, $j = 1, \ldots, m$. The optimization variables, including the total number of waypoints, the locations of each waypoint, and the linear and rotational velocities, are all coupled. Because the waypoints define the location of the path, it follows that the distance and energy distribution along the path can be determined from these waypoints. The number of waypoints, n, and their locations, $W_i(x_i, y_i)$, $i = 1, \ldots, n$, are the dominant

variables, and play the most significant roles when determining the performance index and the amount of recoverable energy along the path. Although the linear and angular speed along each segment also contribute to the performance index, their roles in contribution are generally not dominant in this specific problem, and can be optimized later to further improve the performance along each segment after the waypoints are determined.

To demonstrate this concept, the Balkcom-Mason curve motion planning problem posed in equation 2.15 from Section 2.3 is solved with two types of PSO algorithms. The first of these methods is the modified PSO, discussed in Section 2.5, which solves the design variables all at once (AAO). The second method is the cascaded optimization technique propose above (CPSO), which first optimizes the locations of the waypoints via a heuristic and then optimizes the vehicle's velocities. A comparison of algorithms' resulting performance and computation times are shown below in Fig. 3.2. Examining these results reveals that the CPSO produces results with comparable performance but significantly reduced computation time, and proves that the variables may be solved sequentially.

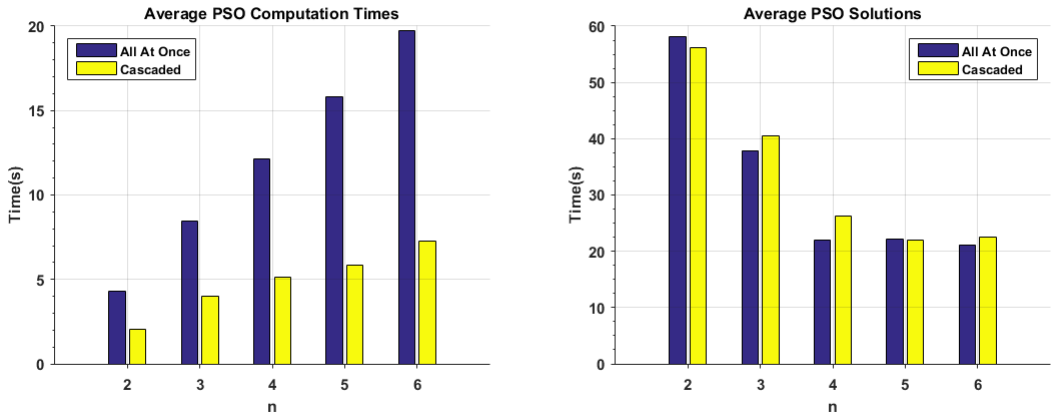

Figure 3.2 PSO vs. CPSO performance and computation times

To solve the energy-constrained Persistent Traveling Vehicle problem, a cascaded optimization algorithm is proposed to cascade the optimization into three distinct steps, wherein the two set of variables will be determined sequentially. We will first examine each of the possible connections between the objectives O, and if appropriate, inject waypoints between two ob-

jective points to allow the vehicle pass through nearby regions with higher energy densities. Secondly, a group of these paths connecting all of the objectives will be selected to allow the UGV to obtain the most amount of energy while traveling least amount of distance, yielding n and $W_i(x_i, y_i)$, $i = 1, \ldots, n$. Finally, we will optimize V_i, $i = 1, \ldots, n$ and ω_i, $i = 1, \ldots, n+1$, to complete the resulting path in minimum time while satisfying net energy gain constraint.

The first step is to determine where the UGV would benefit from detouring from a straight-line path between any two objective points. A connectivity matrix $B_C \in \boldsymbol{R}^{m \times m}$ is generated, where an element $B_C\{i, j\} = 1$ indicates that there is a connecting path between objective point i and j, $i, j = 1, \ldots, m$, $i \neq j$, otherwise, $B_C\{i, j\} = 0$. For each possible connection, we sample the energy density R_{in} of s equidistantly spaced points within a polygon whose vertices are defined by $O_i(x, y + \Delta h)$, $O_i(x, y - \Delta h)$, $O_{i+1}(x, y + \Delta h)$, and $O_{i+1}(x, y - \Delta h)$, where Δh is a user-defined variable which constrains the exploration around the straight-line path between two objective points. We then calculate the standard deviation of the s number of R_{in} values.

If the standard deviation is below a certain threshold, δ_{SD}, we may assume the insolation around the straight-line path between the two objectives to be relatively uniform, and it is not worth diverting from a straight-line path. Subsequently, a standard deviation equal to or greater than δ_{SD} is indicative of a concentrated amount of solar radiation somewhere near the path which would yield additional harvestable energy. In this case, we begin a sub-optimization process which injects two new waypoints, W_1 and W_2, in-between O_i and O_{i+1}, as shown in Fig. 3.3. This brief optimization subprocess utilizes a PSO which employs a heuristic performance index that attempts to produce a path which both passes through energy-dense regions but minimizes deviations from the straight-line path:

$$\begin{aligned} & \min_{\mathbf{X}} -\textstyle\sum_{i=1}^{3} \overline{E}_{in}(i) \cos^2(|\theta_p(i) - \theta_g(i)|) \\ s.t. \quad & O_i \leq W_1 \leq O_{i+1} \\ & O_i \leq W_2 \leq O_{i+1} \\ & W_1 \leq W_2, \end{aligned} \tag{3.7}$$

where the variable set to be optimized, $\mathbf{X}$, contains the x and y locations of the new

waypoints W_1 and W_2, $\overline{E}_{in}(i)$ represents the average ambient energy in along the line segment i, while $\theta_p(i)$ and $\theta_g(i)$ represent the angle of the path and the angle towards $O(i,2)$, respectively. For this brief PSO, each j particle P consists of the candidate locations of W_1 and W_2:

$$P^j = [W_1^j, W_2^j] = [x_1^j, y_1^j, x_2^j, y_2^j]$$

The criterion to determine if the waypoints are required and where they are allocated will apply to each element in B_C with $B_C\{i,j\} = 1$, $i, j = 1, \ldots, m$, $i \neq j$. The average energy per unit distance of the path between $O(i,1)$ and $O(i,2)$ is then saved, regardless of whether it not it was optimized to include W_1 and W_2.

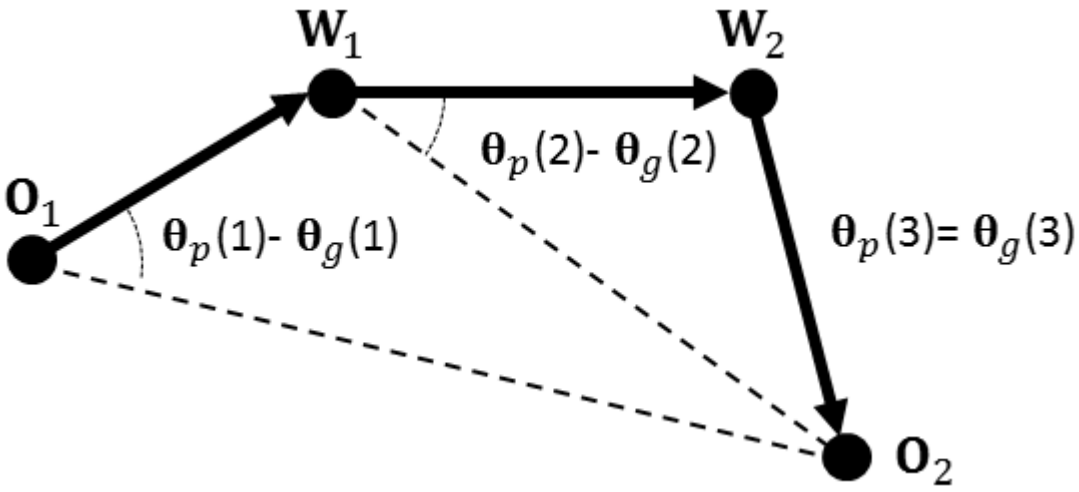

Figure 3.3 Example of deviated path between two objective points

Once the average energy density for each possible connection between the objective points has been recorded, the next step is to determine what route the UGV should take. This step of the optimization process may be solved similarly to the traditional TSP, which can formulated as integer linear programming problem and solved via an existing linear programming solver (C.E. Miller and Zemlin (1960)). However, the performance index for this optimization is different from the classical TSP's, which minimizes the overall distance. Instead, the objective for the persistent motion planning problem here is to maximize the amount of energy per unit

distance, expressed as

$$\max \sum_{i=1}^{m} \frac{\overline{E}_{in}(i)}{d} \tag{3.8}$$

where d is the length of the entire path. The specific objective function above, instead of the distance, is selected here to meet the problem's strict net energy gain constraint. Because the speed at which the vehicle may travel is influenced directly by the energy it may harvest, a path which offers a greater energy intake will allow the vehicle to travel more quickly than an otherwise shorter path with less energy exposure. Therefore, a "good" path for this mission will be one which has (comparatively) short distance and simultaneously has high energy density along the path. Though this step may be trivial for a traveling mission with small number of objective points, the options for the visiting sequence will be a exponential function of the number of objectives.

The final step is to determine the remaining variables, the linear and rotational speed along each segment, which can again be accomplished via the modified PSO algorithm from Chapter 2. A new series of j particles are pseudo randomly initialized in the form of

$$P^j = [V^j(1), \ldots, V^j(n), \omega^j(1), \ldots, \omega^j(n+1)]. \tag{3.9}$$

The initialized particles are constrained by their physical limitations, $V_{min} < V^j < V_{max}$ and $\omega_{min} < \omega^j < \omega_{max}$, and other constraints specified in (2.15). As the particle swarm iterates over time, the particles will eventually converge to an increasingly optimal solution. In the event that none of the initial particles satisfy the net energy constraint, the swarm will be reinitialized until at least one feasible candidate is identified.

CHAPTER 4. HARDWARE AND ENVIRONMENT CHARACTERIZATION

In order to verify the mission planning methods discussed in chapters 2 and 3, it was necessary to construct a time-invariant testing environment and solar-powered mobile robot, as well as to develop an expedient way to characterize the environment's insolation. The development of the test environment is discussed in Section 4.1, while Section 4.2 contains a description of the mobile robot's design. Finally, a method to swiftly characterize the test environment's insolation is presented in Section 4.3.

4.1 Test Environment

Solar panels generate their advertised power yield under a solar radiation density of 1 kW/m^2, which is the approximate value produced during a bright, cloudless day (Garg and Garg (1983)). To simulate this sunlight within the confines of a research lab located underground, we employ high pressure sodium lights, which are commonly used in industrial agriculture greenhouses, shown in Fig. 4.1. The high-pressure sodium lights produce a maximum energy density of 0.6 kW/m^2 underneath the lamp, approximately half of the average outdoor solar radiation. Subsequently, a UGV which can successfully operate in this environment can be expected to achieve higher performance in outdoor tests during sunny or partly cloudy days. The laboratory housing this test environment is equipped with a Vicon Bonita camera system, which can capture the motion of our demonstration robot in a volume of $14 \times 30 \times 12$ ft^3. Because the high-pressure sodium-lights produce 780 nm light, which is identical to the wavelength used by the Bonita's strobe lights, we employ floor tiles coated in blackboard paint to reduce interference between them and the Vicon system (Takashi (1973)).

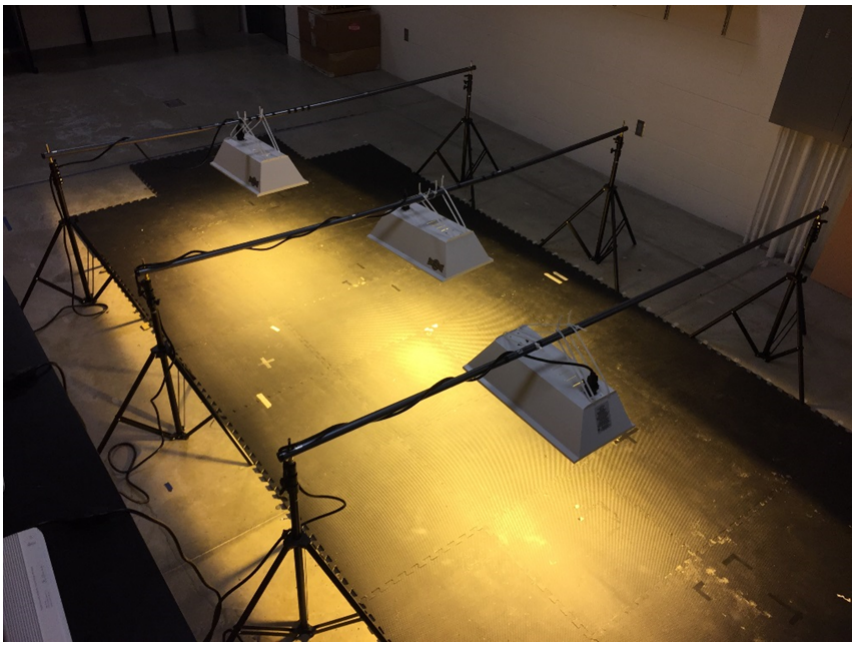

Figure 4.1 Indoor Test Environment.

4.2 Unmanned Ground Vehicle

For a vehicle to be capable of both executing and validating the prescribed motion plan, it must be capable of effectively gathering ambient solar energy and converting it to the operating voltage level, while also measuring the incoming and outgoing energy of the system. Our unmanned ground vehicle, pictured in Fig. 4.2, is based on a Dagu 5 robot chassis controlled by an Arduino Uno and a wireless modem. The ambient solar energy is gathered through a top-mounted solar panel capable of generating 18W of power from bright sunlight, i.e., a solar radiation density of 1 kW/m^2 Recall that the efficiency of a given solar panel is characterized by a nonlinear current-voltage curve, and for this reason the resistance which yields the maximum power point of a solar panel varies. Subsequently, any energy gathered by the UGV's solar panel is directed to an on-board Maximum Power Point Tracking (MPPT) solar charger, which adjusts its resistance to the solar panel in order to produce the optimal load. The MPPT may then pass the energy on to either the vehicle's on-board battery or load (on-board computer, motors, etc). Recovered ambient energy is first delegated to the vehicle's load, and any remaining energy is then used to charge the battery. In the event that the recovered ambient

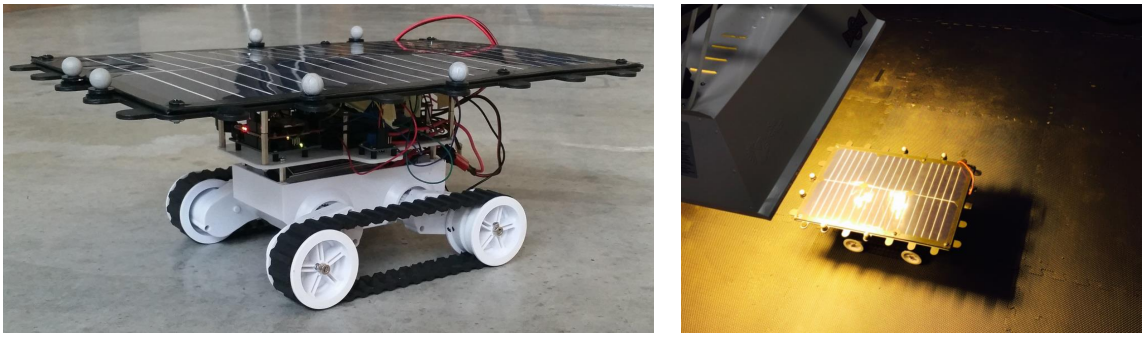

Figure 4.2 Demonstration vehicle with 18W solar panel.

energy is insufficient to supply the required load power, the difference is supplied by the battery. Two voltage/current (V/C) sensors are placed on either side of the MPPT's outgoing lines, as shown in Fig. 4.3. These sensors monitor the vehicle's power flow and provide experimental data in real time.

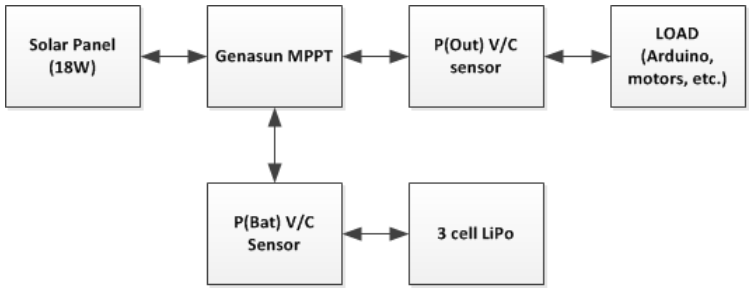

Figure 4.3 Vehicle power system block diagram.

The linear and angular speeds of the UGV are controlled by adjusting the pulse-width modulation (PWM) of the voltages supplied to the motors. For straight line motion, the PWM of both drive-wheels are set the same number. For zero radius turns, the PWMs are set to the same magnitude in opposite directions, and for a circular-arc turn, the interior drive-wheel's PWM is held constant at a predefined value while PWM of the exterior drive-wheel is adjusted to achieve the desired motion. Polynomial functions of the linear speed, angular speed, and power consumed in terms of PWM are interpolated from large amount of sampled data and illustrated in Fig. 4.4.

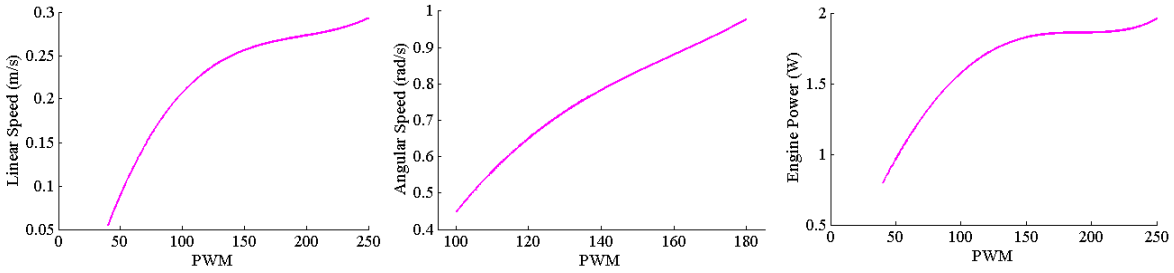

Figure 4.4 Polynomial functions of UGV performance

4.3 Insolation Characterization

Recall that insolation patterns are often highly sporadic, and can not be accurately modeled with continuous functions. To more effectively evaluate the solar radiation map, a scalar field interpolation, $R_{in}(x, y)$ is constructed from a number of discrete solar energy samples to represent the solar energy distribution for a given area. Though these samples can gathered individually by a search vehicle, doing so is inefficient and not be feasible for time-variant environments outside of a laboratory.

Recent work (Kaplan et al. (2016)) has explored the development of insolation maps from simple visual-spectrum imagery. Because photovoltaics utilize energy from several spectrums, including visual, it is possible to correlate the amount of recoverable energy to the amount of visible light in an area.

This characterization is started by first obtaining a high quality image of the environment, which is followed by projecting the image's perspective view on a two-dimensional plane through a geometric spatial transformation. The transformation converts the original geometry data, $I(u, v)$, obtained from the perspective view, into a two dimensional top-view coordinates, $J(x, y)$, via the projective transformation matrix T, expressed as,

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = T \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \tag{4.1}$$

where $T \in \mathbb{R}^{3\times3}$. Constructing the transformation matrix T requires one of two conditions, either the precise height and angle of the camera used to take the picture are given, or the

image contains relative relationship between two reference frames, I and J. For example, the indoor test environment discussed above utilizes $60 \times 60\, cm^2$ floor tiles. Each of the four corners of a tile can be identified as one of the elements in $I(u, v)$, and its corresponding coordinates in $J(x, y)$ can be determined by setting up a two-dimensional reference frame in the real world. The transformation matrix T can be solved via (4.1) when sufficient number of elements in both I and J frames are identified. The example in Figure 4.5 uses four identified points to obtain the transformation matrix, where each pixel of $J(x, y)$ represents 1 cm^2. In cases where a camera is affixed at a given location and orientation on a UGV, the transformation matrix T can be determined from the initial setting and can then be used for geometry transformation of any subsequent images.

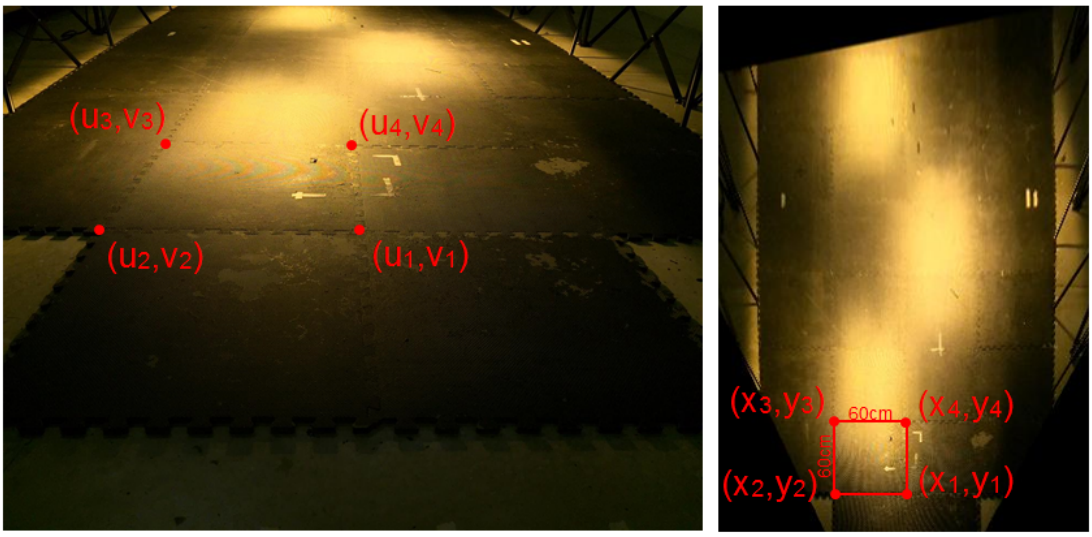

Figure 4.5 Projective Transformation of Test Environment

Once a top-down view of the environment has been generated, the next step is to determine the amount of available energy at each location. Because photovoltaic cells operate under the visual light spectrum, among others, the available energy within the environment should correspond to the brightness of the image. To find the relationship between the solar radiation and the brightness values, the image is converted to greyscale to generate a double-precision matrix with indices representing the locations and elements corresponding the brightness at these locations.

By sampling the energy density R_{in} at two separate points which demonstrate a distinct difference in brightness within the test environment, a linear relationship between the sampled power and the brightness value may be expressed as

$$\begin{aligned} R_{in}^{h} &= a_1 B^{h} + a_2 \\ R_{in}^{l} &= a_1 B^{l} + a_2 \end{aligned} \tag{4.2}$$

where a_1 and a_2 are coefficients to be determined, R_{in}^{h} and R_{in}^{l} are sampled energy density at corresponding high brightness B^{h} and low brightness B^{l} locations. This linear relationship allows us to predict the energy density for any given brightness value B within the image. Certain environments may require more complex scaling functions, which can be generated in a similar fashion. With the geometric transformation and brightness conversion, an insolation map, $R_{in}(x, y)$, is generated from the captured image, and is utilized to provide energy distribution information for a scalar field interpolation, which may then be used for mission planning.

To provide a comparison, a polynomial surface fit of 5th order in both x and y was applied data sampled from the test environment, and shown below in Figs. 4.6 and 4.7. This high-degree polynomial surface of $R^2 = 0.6595$ is clearly a poor fit for the scattered data.

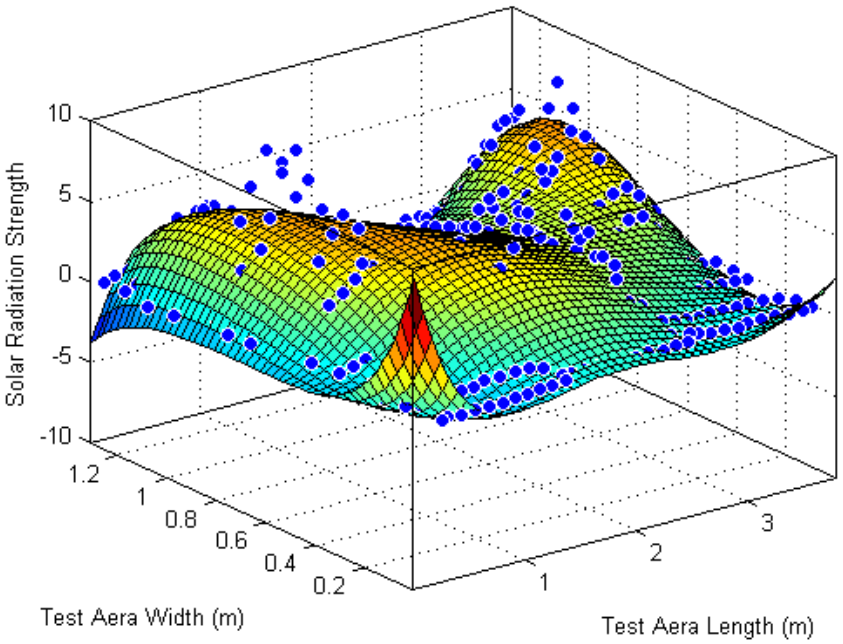

Figure 4.6 3D Polynomial Surface Fit of Insolation Data

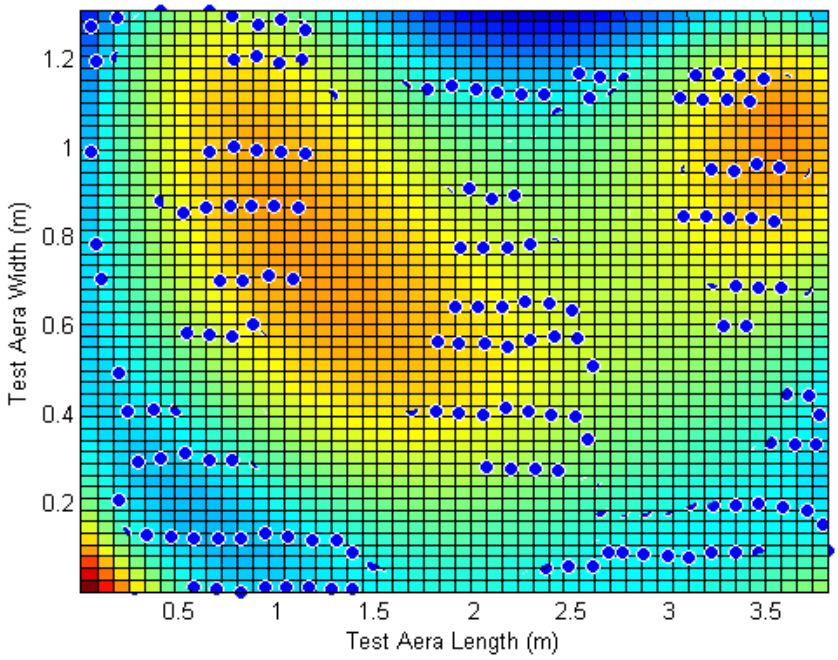

Figure 4.7 2D Polynomial Surface Fit of Insolation Data

CHAPTER 5. SIMULATION AND EXPERIMENT RESULTS

This Chapter contains the simulated and experimental results for the energy-constrained Transit Problem posed in Chapter 2, and the energy-constrained Persistent Traveling Vehicle problem posed in Chapter 3. Section 5.1 begins with a simulated environment, and includes solutions for a naive straight-line, a Balkcom-Mason, and a Pseudo-Dubins solution to the transit problem, as well as a comparison case using a conventional path planning algorithm. Section 5.2 presents experimental data for the transit problem using naive straight-line, Balkcom-Mason, and Pseudo-Dubins curves. A conventional TSP-approach is applied to the persistent traveling vehicle problem in Section 5.3, while Section 5.4 concludes with experimental results of a solution to the persistent traveling vehicle problem generated from the CPSO algorithm.

Each of the plans presented below has three power schedule parameters which characterize the UGV's performance: the anticipated intake power, P_{IN} (—), the consumed power, P_{OUT} (—), and the change of battery energy, $\Delta Battery$ (—). In the cases where experimental data was gathered and compared to predicted trends, the corresponding experimental data is denoted as P_{IN} (- - -), P_{OUT} (- - -), and $\Delta Battery$ (- - -). Please note that energy gained by the system is denoted as positive, while energy expended from the system is negative.

5.1 Transit Problem Simulation Results

We now examine a scenario where our vehicle is tasked with crossing a simulated $5.5 \times 5.5\ m^2$ sparse-energy area which yields a maximum power intake of less than 7W. Results for a naive straight-line solution, a Balkcom-Mason solution, a Pseudo-Dubins solution, and a conventional Bellman-Ford graph search solution are presented and discussed below.

5.1.1 Straight-Line Solution

A simple straight-line solution between our starting and ending location, using the greatest possible uniform-velocity which can satisfy our energy constraint, is pictured below in Fig. 5.1 and Fig. 5.2. This solution requires a total travel time of 32.27 seconds and yields a net energy gain of 0.43 J.

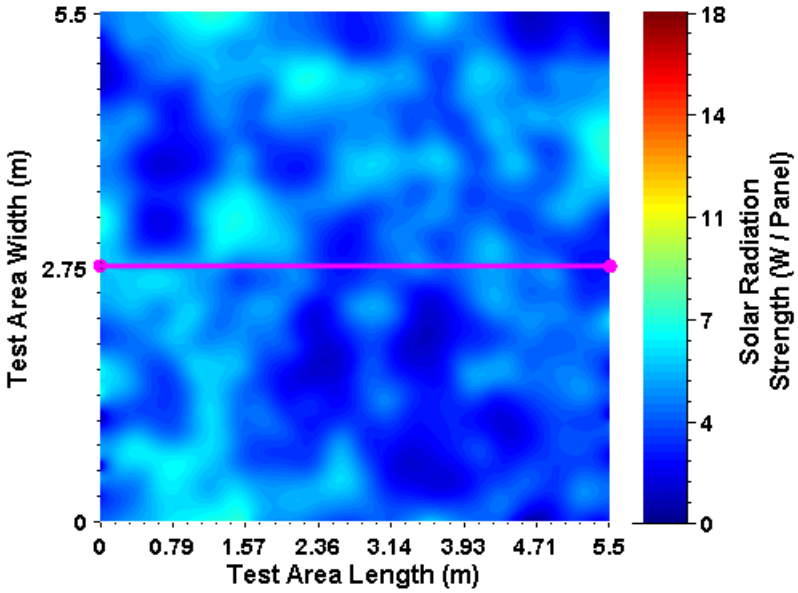

Figure 5.1 Straight-Line Path for Virtual Scenario

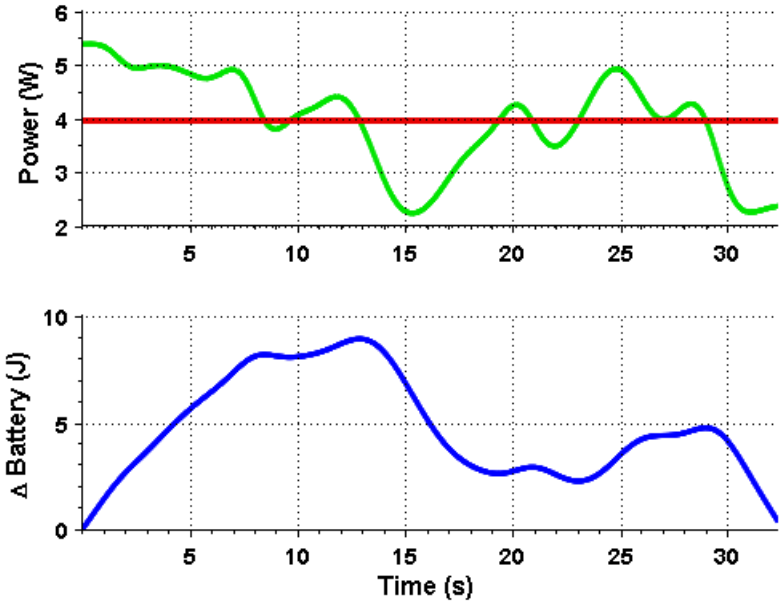

Figure 5.2 Simulated Straight-Line Power & Energy Schedule

5.1.2 Balkcom-Mason Solution

We then generate a 5-waypoint Balkcom-Mason motion plan, presented below in Figs. 5.3 and 5.4. This plan requires the vehicle to move for 26.86 seconds and predicts a 1.05 J net energy gain.

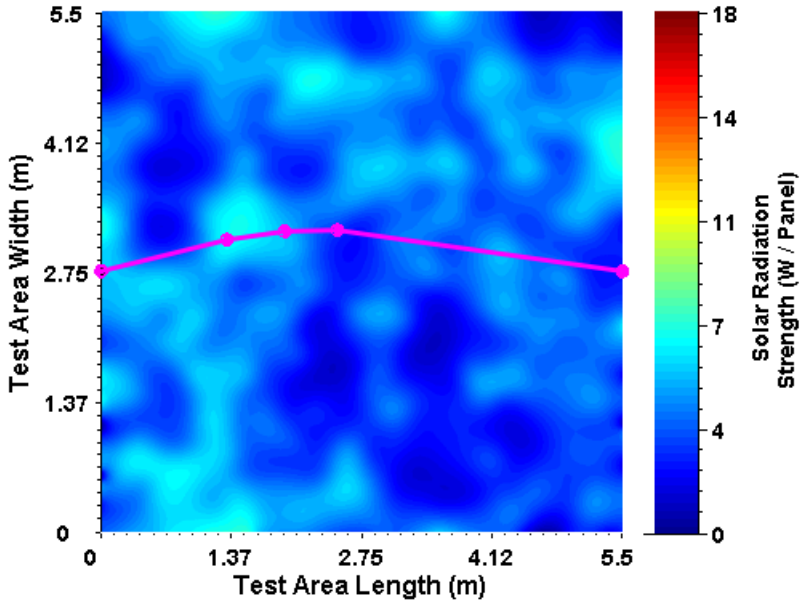

Figure 5.3 Balkcom-Mason Path for Virtual Scenario

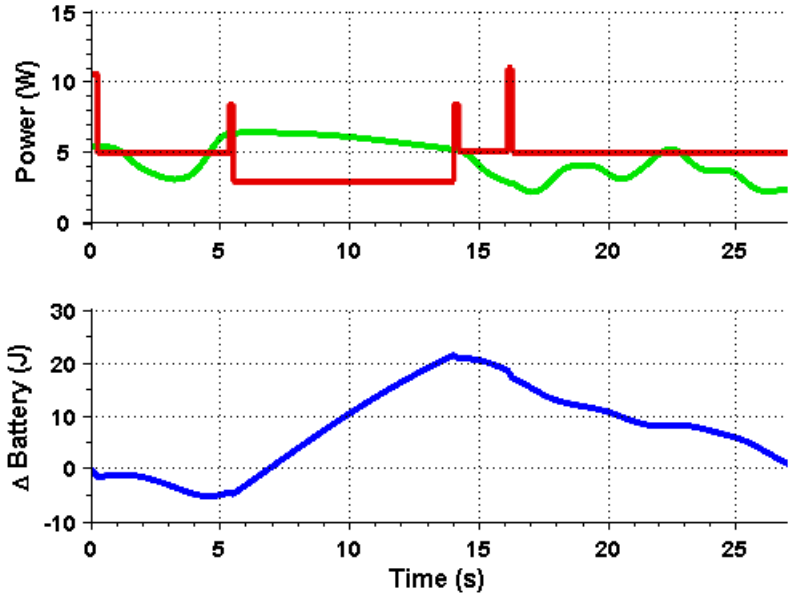

Figure 5.4 Simulated Balkcom-Mason Power & Energy Schedule

5.1.3 Pseudo-Dubins Solution

Next, we generate a 5-waypoint Pseudo-Dubins motion plan, summarized below in Fig. 5.5 and Fig. 5.6. The Pseudo-Dubins solution predicts a 26.03 second travel time accompanied by a 0.26 J net energy increase.

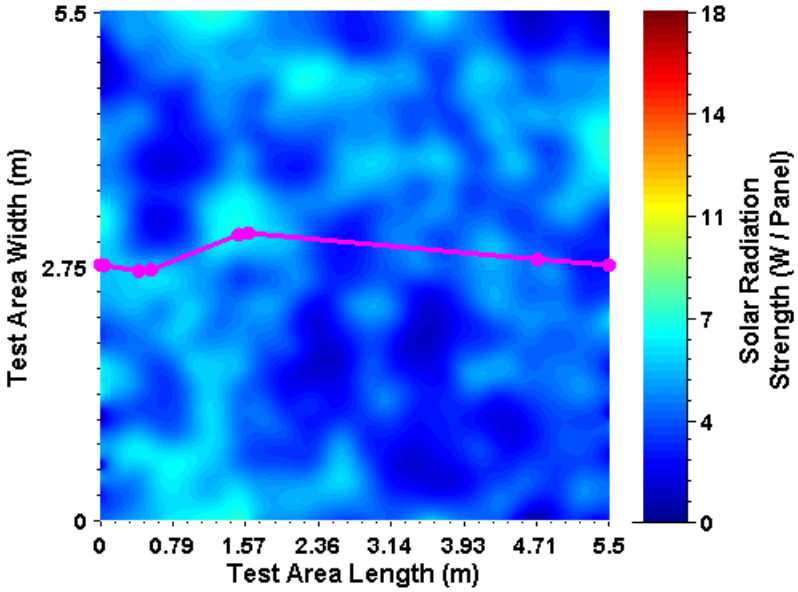

Figure 5.5 Pseudo-Dubins Path for Virtual Scenario

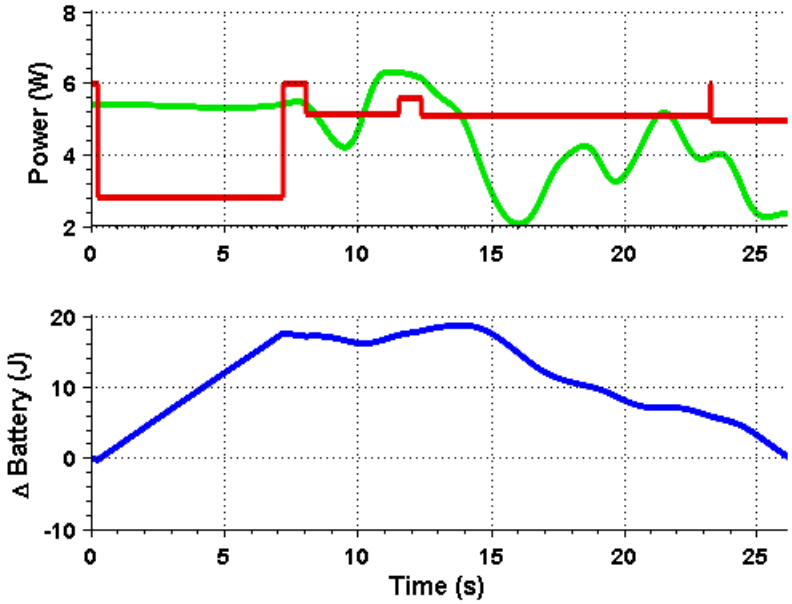

Figure 5.6 Simulated Pseudo-Dubins Power & Energy Schedule

While both the Balkcom-Mason and Pseudo-Dubins plans offer a significant time reduction from the naive straight-line solution (16.76% and 19.34%, respectively), the Pseudo-Dubins offers only a 3.09% reduction from the Balkcom-Mason motion plan.

5.1.4 Bellman-Ford Solution

To demonstrate the advantages of the modified PSO over conventional motion-planning methods in solving this type of problem, we conclude by examining a motion plan generated from the Bellman-Ford graph search algorithm (Bellman (1958)). The Bellman-Ford algorithm, like other conventional motion planners discussed in Chapter 1, works by identifying the optimal path comprised of multiple possible path segments. In order to adapt the classical Bellman-Ford algorithm technique to our problem, the same simulated area is divided into a set of grids and path segments, which are denoted by the edges connecting any node (labeled as circles in Fig. 5.7) in a given column to any other node in the adjacent columns.

The "cost" of each segment is determined by the minimum amount of time required for the vehicle to travel between one node to another while avoiding a negative net energy change. Because the Bellman-Ford algorithm identifies some optimal assortment of edges, compliance with the Transit Problem's net energy constraint mandates that each of these edges individually meet the net energy constraint. For any edge that cannot satisfy the net energy gain constraint, a relatively large cost is assigned so as to render it infeasible to the algorithm. The Bellman-Ford algorithm (Bellman (1958)) is used to search for the minimum cost path from the initial point to the final point. Grids ranging from $5x5$ to $25x25$ nodes were examined, and the best performing motion plan identified by the Bellman-Ford method is presented below in Fig. 5.7 and Fig. 5.8. This plan requires a 38.52 second travel time, and produces a net energy increase of 0.13 J.

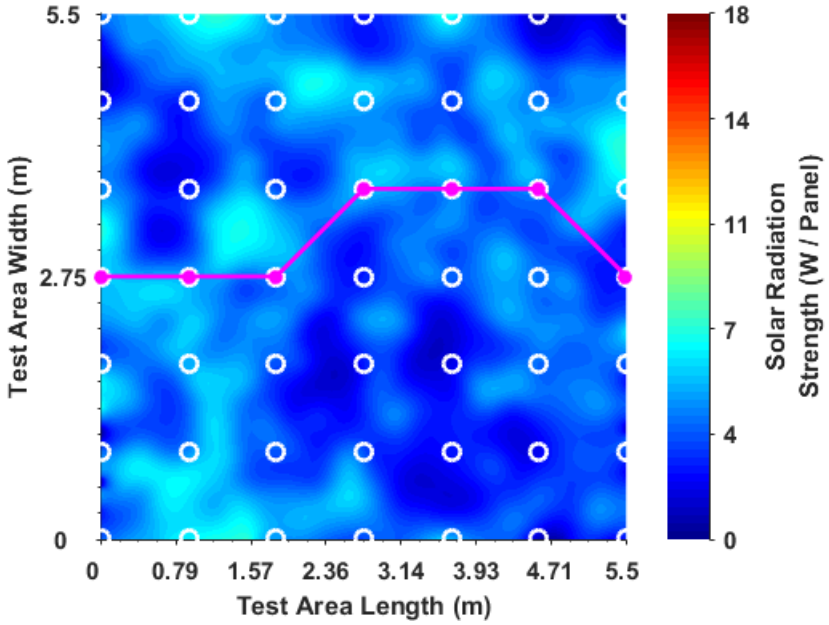

Figure 5.7 Path from Bellman-Ford for Virtual Scenario

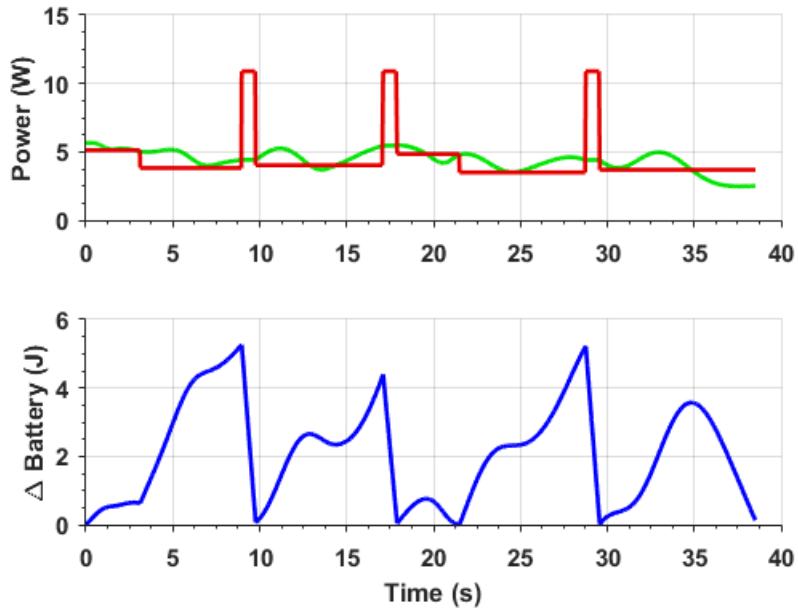

Figure 5.8 Simulated Power & Energy Schedule from Bellman-Ford Method

The Bellman-Ford plan is, unsurprisingly, slower than the Balkcom-Mason (43.41%) and Pseudo-Dubins (46.50%), yet also slower than the naive straight-line solution (22.90%) due to the constraint on each path segment. It is clear from this method that adapting conventional motion planners to this type of problem cannot yield result with improved performance.

5.2 Transit Problem Experiment Results

We now employ the test environment described in Chapter 4 to generate new plans and gather experimental data. Three high-pressure sodium lights are distributed within a 3.8×1.31 m^2 area. The resulting solar energy density distribution is mapped and used to generate a naive straight-line solution, a 5-waypoint Balkcom-Mason solution, and a 5-waypoint Pseudo-Dubins solution. The resulting motion plans were executed by the demonstration vehicle five separate times and are presented below.

5.2.1 Straight-Line Solution

The naive straight-line solution, including path, predicted variable history, and the recorded experimental data, is shown in Figs. 5.9-5.10. The anticipated travel time for the straight-line solution is 57.71 seconds with a net energy increase of 1.19 J, while the average experimental travel time is 57.13 seconds with a minimum of 56.42 seconds and a maximum of 57.79 seconds, and an average experimental net energy increase 1.89 J with a minimum net energy increase of -0.17 J and a maximum of 3.97 J. The average standard deviation of the experimental ΔBattery from the prediction is 1.68 J, with a minimum of 1.46 J and a maximum of 1.97 J.

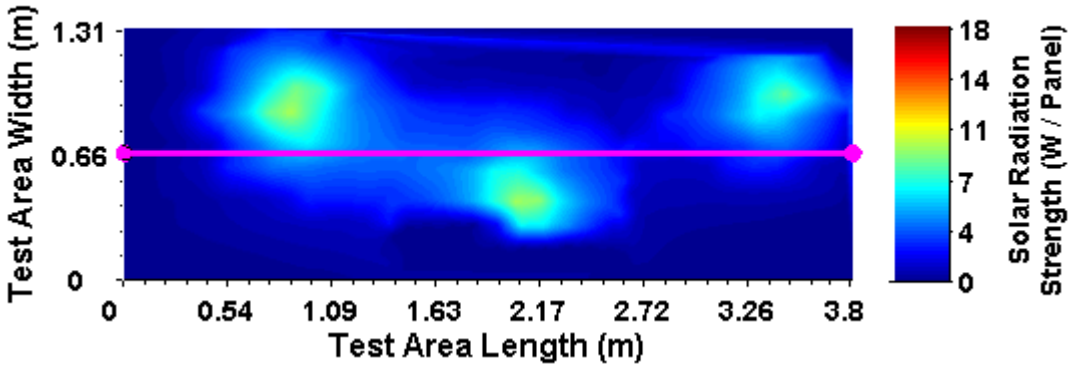

Figure 5.9 Straight-Line Path for Experimental Environment

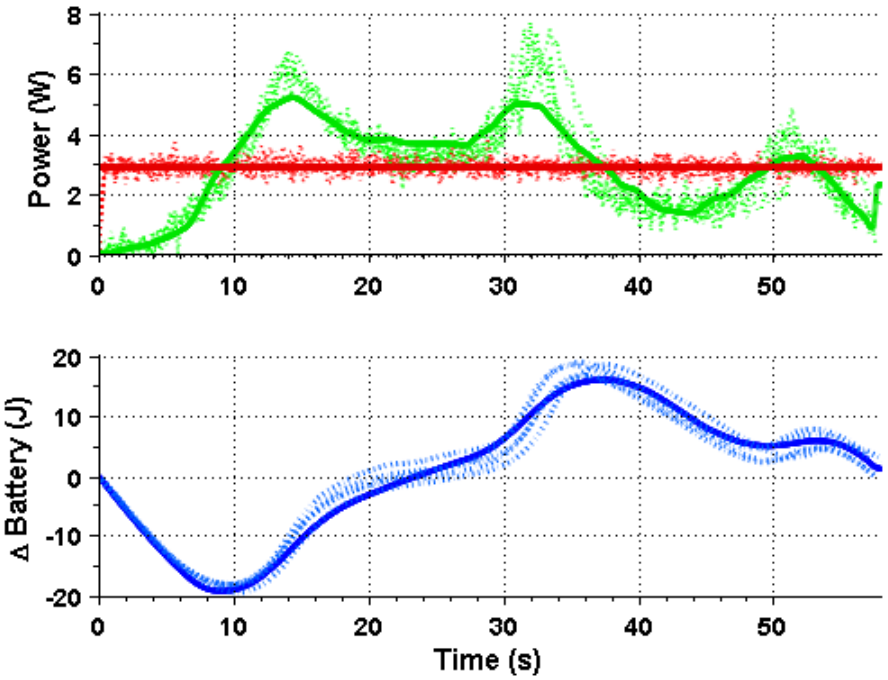

Figure 5.10 Experimental Straight-Line Power & Energy Data

The straight-line solution requires the vehicle to travel at the speed of 0.066 m/s, which is very close to the UGV's minimum speed of 0.055 m/s. Even if a straight-line path from start to finish were broken up into smaller line segments with nonuniform velocities, the experimental data above demonstrates that the vehicle would be unable to travel any (significantly) slower through the high-energy regions, and thus would have insufficient energy to travel more quickly in the sparse regions. A straight line, in this case, cannot yield a swift route from start to finish while satisfying the net energy gain constraint.

5.2.2 Balkcom-Mason Solution

The path and predicted variable history of the Balkcom-Mason solution can be found in Figs. 5.11-5.12. The anticipated travel time for the straight-line solution was 25.01 seconds with a net energy increase of 0.25 J, while the average experimental travel time was 24.78 seconds with a minimum of 24.41 seconds and a maximum of 25.25 seconds, and an average experimental net energy increase -0.03 J with a minimum net energy increase of -1.30 J and a maximum of 1.72 J. The average standard deviation of the experimental ΔBattery from the prediction is 1.48 J, with a minimum of 0.92 J and a maximum of 1.96 J.

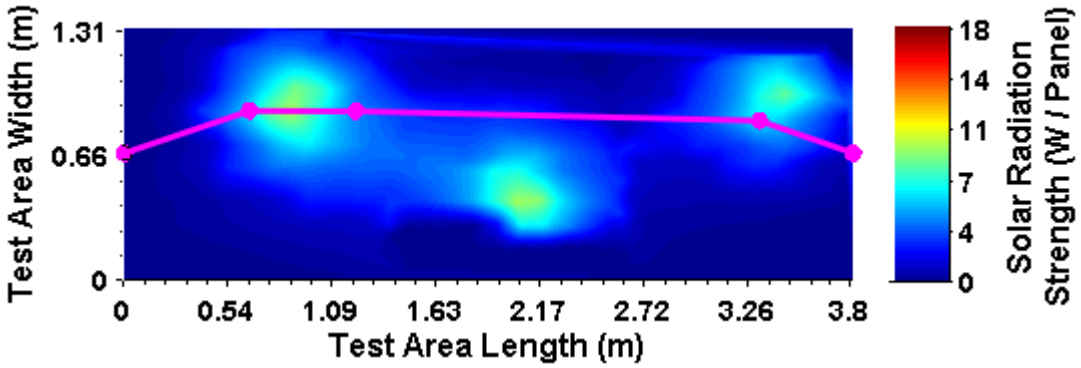

Figure 5.11 Balkcom-Mason Path for Experimental Environment

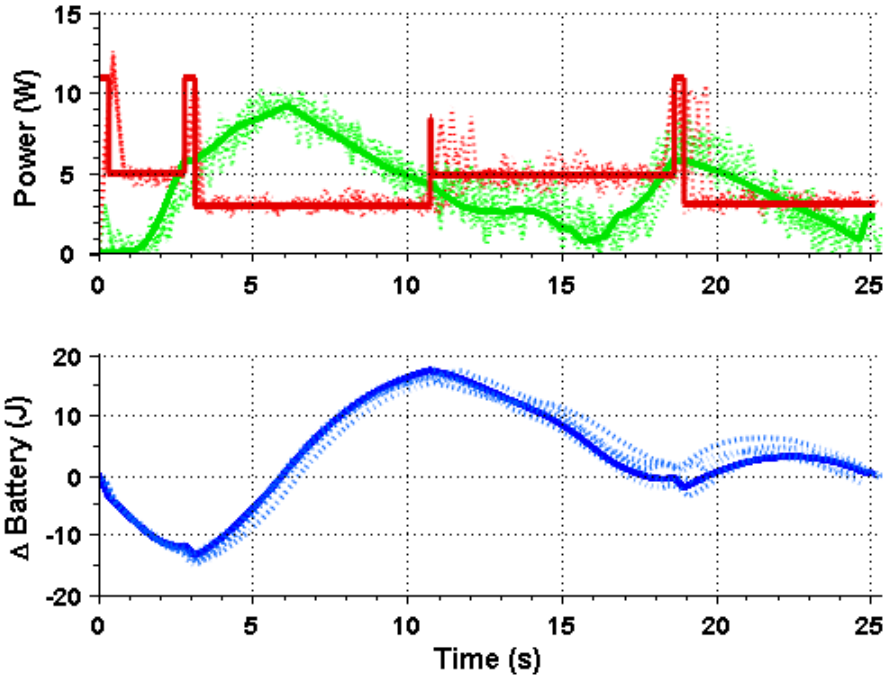

Figure 5.12 Experimental Balkcom-Mason Power & Energy Data

5.2.3 Pseudo-Dubins Solution

The pseudo-Dubins solution, including its path, predicted variable history, and the recorded experimental data, are presented in Figs. 5.13-5.14. The anticipated travel time for the pseudo-Dubins solution is 20.22 seconds with a net energy increase of 3.39 J, while the average experimental travel time is 20.20 seconds with a minimum of 19.90 seconds and a maximum of 20.41

seconds, and an average experimental net energy increase 4.17 J with a minimum net energy increase of 1.53 J and a maximum of 7.99 J. The average standard deviation of the experimental ΔBattery from the prediction is 1.66 J, with a minimum of 1.13 J and a maximum of 2.24 J.

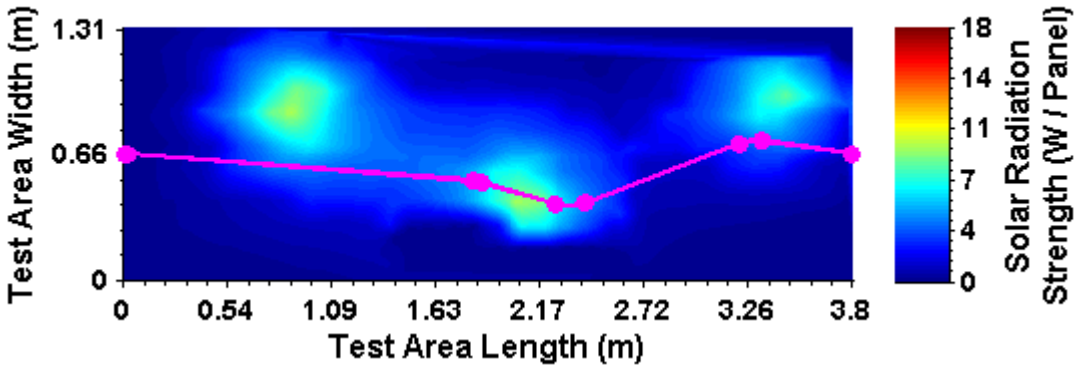

Figure 5.13 Pseudo-Dubins Path for Experimental Environment

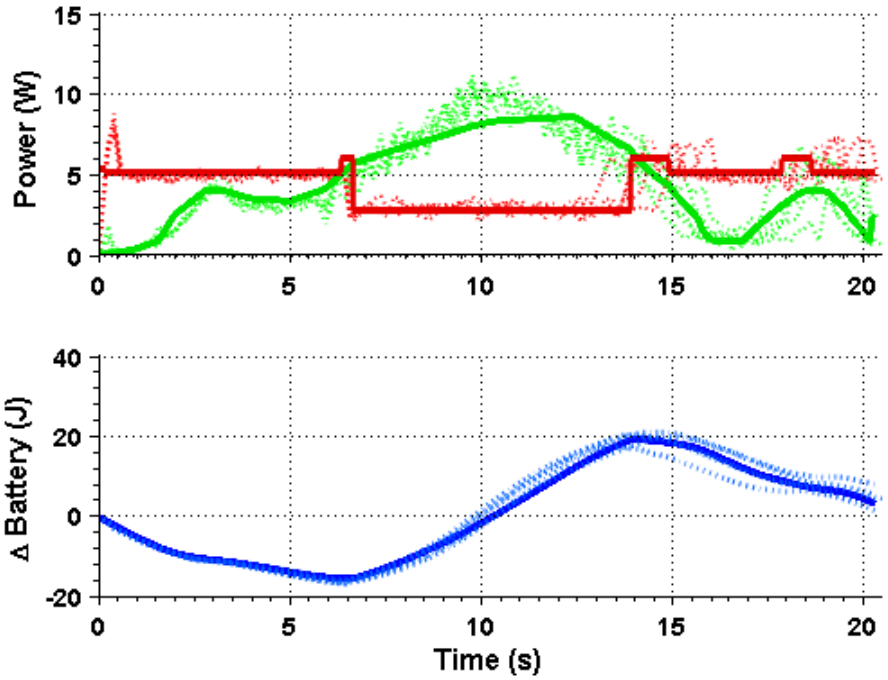

Figure 5.14 Experimental Pseudo-Dubins Power & Energy Data

Again, both the Balkcom-Mason and Pseudo-Dubins plans resulted in decreased transit time from the naive straight-line case (56.66% & 64.97% predicted and 56.59% & 64.63% average experimental, respectively). More significantly, the Pseudo-Dubins solution constituted a

marked improvement from the Balkcom-Mason plan, offering predicted travel time reduction of 19.16% and an average experimental time reduction of 18.54%. The Balkcom-Mason solution requires that the UGV moves slowly underneath the leftmost energy source and pass close to the rightmost source in order to meet the energy constraint. In contrast, the Pseudo-Dubins solution was planned to gather sufficient energy from the middle source alone, and consumes less energy to complete its turns by preserving some of the vehicle's translational motion towards the goal during the turn. The small discrepancies observed between the anticipated and experimental energy trends may be contributed by small deviations in the vehicle's path from its planned path, due to interference between our test environment's motion capture system and the infrared light emitted by the high-pressure sodium lamps. In testing, the demonstration vehicle averaged a 4.93% error in distance traveled and 2.27% error in angle turned. Measurement noise in the UGV's voltage/current sensors may also contribute to these small experimental discrepancies.

5.3 Persistent Traveling Vehicle Problem Simulation Results

We now begin our analysis of Persistent Traveling Vehicle problem introduced in Chapter 3. Four objective points within the test environment were selected and placed within the test environment. The UGV is tasked with starting at objective point 1, traveling to objective points 2-4 in some order, and returning to its starting position and orientation as quickly as possible while ensuring a nonnegative net energy change. To benchmark the performance of the planned motion using the CPSO method discussed in Chapter 3, we first provide a simulated motion plan obtained from the classical TSP solution. Different from the CPSO method, the classical TSP method does not consider the environment's energy distribution when searching for the optimal path. Instead, it seeks to minimize the vehicle's travel time by finding the shortest distance between the objectives, but allows the vehicle to slow down at each path segment to meet the net energy gain constraint. The comparative planned path and its energy schedules are shown in Figs. 5.15-5.16. The TSP plan results in a vehicle travel time of 43.82 seconds, and a final $\Delta Battery$ of 1.187 J.

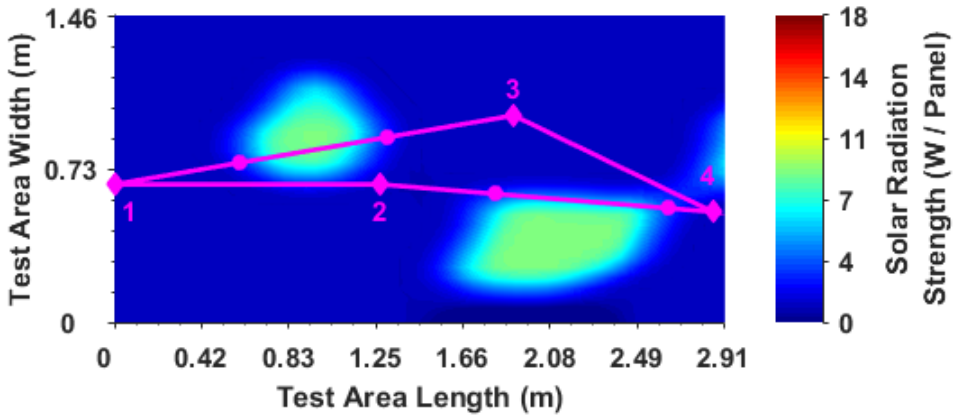

Figure 5.15 Comparison Path Plan

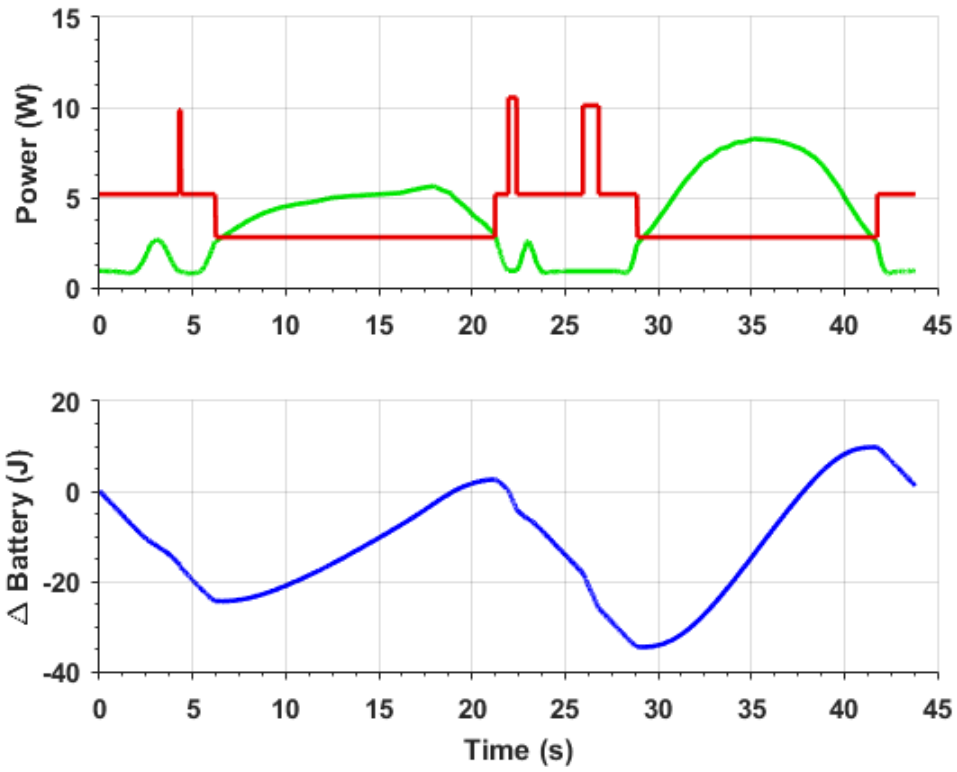

Figure 5.16 Comparison Energy Schedule

5.4 Persistent Traveling Vehicle Problem Experiment Results

Finally, the CPSO method discussed in Chapter 3 is applied to produce an optimized motion plan and energy schedules. The resulting mission plan was executed by the demonstration

vehicle five separate times. The planned path and history of the associated power schedules are shown in Figs. 5.17 & 5.18.

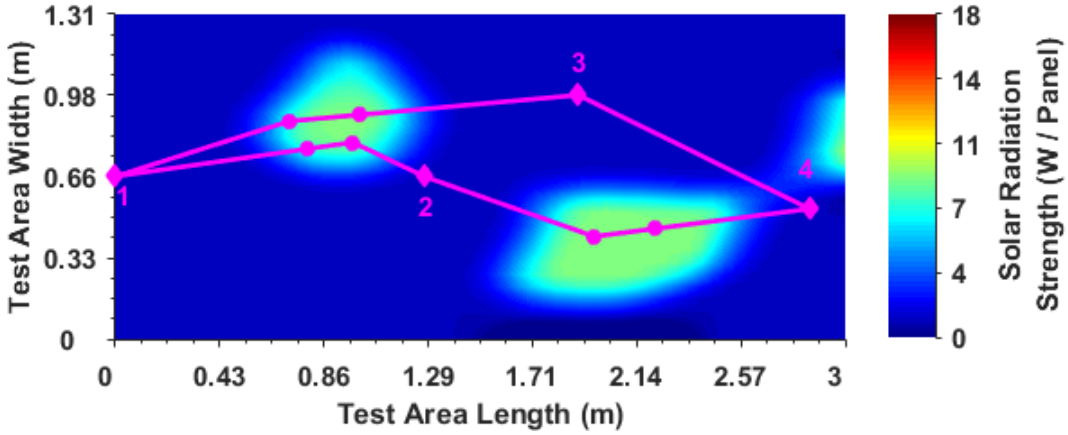

Figure 5.17 Optimal Planned Path

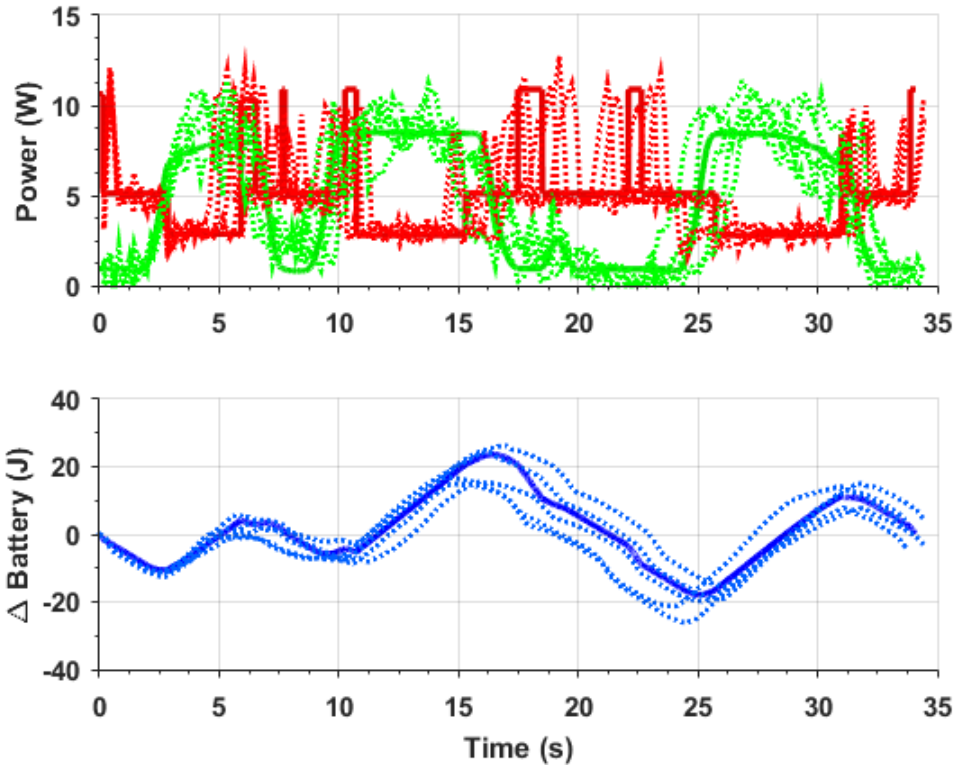

Figure 5.18 Optimized Energy Schedules

The optimized plan predicts a vehicle travel time of 34.09 seconds and a $\Delta Battery$ of 0.0037 J, while the average experimental travel time and $\Delta Battery$ of the five trials were 33.97

s and 0.0024 J, respectively. The average standard deviation of the experimental ΔBattery from the prediction was 4.6034 J, with a minimum of 3.1764 J and a maximum of 5.9566 J. The experimental data gathered was prone to significantly more error than the previous experiments, ddue in no small part to this plan calling for more than double the number of vehicle movements than the previous plans. Even a small error in any one of these movements could bring the vehicle off of its predicted path and cause a propagating series of deviations from the anticipated energy states.

Note that the CPSO algorithm added additional waypoints between the objective point pairs (1,2), (1,3) and (2,4), where the energy density is non-uniformly distributed along the corresponding segment, and that the locations of these inserted waypoints were optimized to slightly deviate from the straight-line segment to harvest more energy along the path. There were no waypoints added between the objective point pair (3,4) as the energy density along the corresponding segment is relatively uniform. Though these additions result in the CPSO path being slightly longer than the conventional TSP solution's, the CPSO plan requires some 22.20% less time to complete.

CHAPTER 6. CONCLUSION

This thesis has explored motion planning techniques for a solar-panel equipped unmanned ground vehicle. Variants of Particle Swarm Optimization were applied to solve transit problems under strict energy constraints, and a Cascaded Particle Swarm Optimizer was introduced to generate a mission plan which required the vehicle to persistently visit a number of objective points. An indoor test environment and custom-built mobile robotics platform were designed and built, and simulated and experimental results were presented which demonstrated the effectiveness of these approaches over conventional methods. Though the methods presented constitute a first step in the relatively new field of insolation-based mission planning for solar-powered vehicles under energy constraints, they already vastly outperform conventional path planning techniques such as the Bellman-Ford graph search.

With the advancement of autonomous technology, unmanned systems have alleviated human operators from numerous tedious tasks where system endurance plays a crucial role. Current technology employed in solar-powered unmanned systems is subject to design and power limitations and varying environments. Intelligently harvesting energy from environments and scheduling power consumption to optimize the desired system performance will significantly improve the vehicle's endurance, and any unmanned vehicle which can access solar energy may potentially benefit from the results of this work.

Future efforts must address the swift and accurate characterization of an environment's insolation, and be proven in time-variant environments of non-uniform terrain, and additional effort will be required to further reduce the computation time required by these methods and allow real-time mission planning.

BIBLIOGRAPHY

Anthony J. Carfang, E. W. F. and Kingston, D. B. (2014). Cascaded optimization of aircraft trajectories for persistent data ferrying. *Journal of Aerospace Information Systems*, 11.

Balkcom, D. J. and Mason, M. T. (2002). Time optimal trajectories for bounded velocity differential drive vehicles. *The International Journal of Robotics Research*, 21(3):199–217.

Bellman, R. (1958). On a routing problem. *Quarterly of Applied Mathematics*, 16:87–90.

Brett Bethke, J. P. H. and Vian, J. (2008). Group helath management of uav teams with applications to persistent surveillance.

C.E. Miller, A. T. and Zemlin, R. (1960). Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM)*.

de J Mateo Sanguino, T. and Gonzalez Ramos, J. E. (2013). Smart host microcontroller for optimal battery charging in a solar-powered robotic vehicle. *Mechatronics, IEEE/ASME Transactions on*, 18(3):1039–1049.

Dubins, L. E. (1957). On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of mathematics*, pages 497–516.

Eberhart, R. C. and Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Proceedings of the sixth international symposium on micro machine and human science*, volume 1, pages 39–43. New York, NY.

Garg, H. and Garg, S. (1983). Prediction of global solar radiation from bright sunshine hours and other meteorological data. *Energy Conversion and Management*, 23(2):113–118.

Glassner, A. S. (1989). *An introduction to ray tracin*. Morgan Kaufmann.

Gupta, S. G., Ghonge, M. M., and Jawandhiya, P. (2013). Review of unmanned aircraft system (uas). *technology*, 2(4).

Heng, I., Zhang, A. S., and Harb, A. (2011). Using solar robotic technology to detect lethal and toxic chemicals. In *IEEE Global Humanitarian Technology Conference*, pages 409–414.

Kaplan, A., Kingry, N., Top, J. V. D., Patel, K., Dai, R., and Grymin, D. (2016). Motion planning for persistent traveling solar-powered unmanned ground vehicles. In *IEEE International Conference on Robotics and Automation (ICRA)*. Submitted 9/15/2015.

Kaplan, A., Kingry, N., Uhing, P., , and Dai, R. (2015a). Time-optimal path planning with power schedules for a solar-powered ground robot. Conditionally Accepted 7/31/2015.

Kaplan, A., Uhing, P., Kingry, N., and Dai, R. (2015b). Integrated path planning and power management for solar-powered unmanned ground vehicles. In *IEEE International Conference on Robotics and Automation (ICRA)*.

Kavraki, L. E., Švestka, P., Latombe, J.-C., and Overmars, M. H. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *Robotics and Automation, IEEE Transactions on*, 12(4):566–580.

Lawrance, N. R. and Sukkarieh, S. (2011). Autonomous exploration of a wind field with a gliding aircraft. *Journal of Guidance, Control, and Dynamics*, 34(3):719–733.

Lever, J. H., Streeter, A., and Ray, L. (2006). Performance of a solar-powered robot for polar instrument networks. In *Robotics and Automation, IEEE International Conference on*, pages 4252–4257.

Marc Raibert, Kevin Blankespoor, G. N. R. P. and the BigDog Team (2008). Bigdog, the rough-terrain quadruped robot. The International Federation of Automatic Control.

Mario Valenti, Daniel Dale, J. H. and de Farias, D. P. (2007). Mission health management for 24/7 persistent surveillance operations. AIAA Guidance, Control, and Navigation Conference.

Mei, Y., Lu, Y.-H., Hu, Y. C., and Lee, C. G. (2004). Energy-efficient motion planning for mobile robots. In *2004 IEEE International Conference on Robotics and Automation*, volume 5, pages 4344–4349.

N. Kemal Ure, Girish Chowdhary, T. T. J. P. H. M. A. V. J. V. (2015). An automated battery management system to enable persistent missions with multiple aerial vehicles. *IEEE Transactions on Mechatronics*, 20.

Nigam, N. (2014). The multiple unmanned air vehicle persistent surveillance problem: A review. *Machines*, pages 13–72.

Plonski, P. A. and Isler, V. (2014). A competitive online algorithm for exploring a solar map. International Conference on Robotics and Automation.

Plonski, P. A., Tokekar, P., and Isler, V. (2013). Energy-efficient path planning for solar-powered mobile robots*. *Journal of Field Robotics*, 30(4):583–601.

Ray, L. E., Lever, J. H., Streeter, A. D., and Price, A. D. (2007). Design and power management of a solar-powered cool robot for polar instrument networks. *Journal of Field Robotics*, 24(7):581–599.

Robert W. Button, John Kamp, T. B. C. J. D. (2009). A survey of missions for unmanned undersea vehicles. Technical report, Rand Corporation.

Sertac Karaman, E. F. (2011). Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research*.

Sulaiman, A., Inambao, F., and Bright, G. (2013). Development of solar hydrogen energy for mobile robots. In *6th IEEE Robotics and Mechatronics Conference*, pages 14–19.

Takashi, S.-I. (1973). Blackening for sub-mm wavelengths. *Infrared Physics*, 13:301–303.

Takayama, L., Ju, W., and Nass, C. (2008). Beyond dirty, dangerous and dull: what everyday people think robots should do. In *Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction*, pages 25–32.

Vasisht, S. and Mesbahi, M. (2015). Trajectory design and coverage control for solar-powered uavs. AIAA Guidance, Navigation, and Control Conference.

Vaussard, F., Rétornaz, P., Liniger, M., and Mondada, F. (2013). The autonomous photovoltaic marxbot. In *Intelligent Autonomous Systems*, pages 175–183. Springer.

Voth, D. (2004). A new generation of military robots. *IEEE Intelligent Systems*.

Wang, T., Wang, B., Wei, H., Cao, Y., Wang, M., and Shao, Z. (2008). Staying-alive and energy-efficient path planning for mobile robots. In *American Control Conference*, pages 868–873.

Zeng, W. and Church, R. (2009). Finding shortest paths on real road networks: the case for a*. *International Journal of Geographical Information Science*, 23(4):531–543.