Constructing a Movement Model for a Small Unit

J. Vince Pulido[1], Craig Lennon[2], Mary Anne Fields[2]
Laura Barnes[1]
[1]Department of Systems and Information Engineering, University of Virginia,
151 Engineer's Way, Charlottesville, VA 22904
[2]Army Research Lab, Vehicle Technology Directorate
28000 Powder Mill Rd. Adelphi, MD 20783

Abstract
The Autonomous Squad Member (ASM) project is a research effort to develop the intelligence that would allow a ground robot to support a dismounted unit with little human direction and intervention. In order to plan actions with respect to current priorities and to detect events which might signal a change in priorities, such intelligence must make predictions of the future actions of the soldiers it supports. We developed a model of squad movement allowing the robot to estimate future positions of the squad and arrival times. Specifically, we developed a technique which employs an iterated A* algorithm to bound the small unit's movement to a specified area based on an a priori map of the terrain, known mission constraints, and a weighted combination of costs, where the weight is derived from a possible mission context. By varying these weights, we can develop a diverse population of paths which represent the predicted position of the unit. We further model the time at which arrival at a given location is likely to occur.

Key Words: A* search, path planning, autonomous vehicles, movement reasoning

1. Introduction

Ground-based robots designed for battlefield deployment will play significant roles in future military missions. While some of these systems are capable of navigating through rough and uncertain terrain, they lack the capability to autonomously engage with and assist human soldiers as teammates within the context of a military mission.

There are a few notable literatures on how a military robot should interact with humans, developed from a human-robot interaction perspective (Barnes et. al. 2010, Parasurman et al. 2007). Additionally, substantial work documents the development of scripted tactical behaviors for military robots to execute, including autonomous tactical navigation between two points (Childers et al. 2010). With respect to traveling as a team, Fields et. al. (2009) presents algorithms and interface concepts that allow soldiers to efficiently interact with a robotic swarm participating in a representative convoy mission. While previous work on military human-robot interaction have given robots the capability to work intelligently for humans *as tools*, there has not been work done to give military robot the ability to work effectively with humans *as teammates*.

The Autonomous Squad Member (ASM) project is a research effort whose goal is to provide a robot with adequate intelligence to autonomously operate within a squad, assisting its team members in accomplishing a mission. The ASM is envisioned to support the team's success by playing roles such as carrying equipment or specialized sensors, providing high power communication antennae, providing emergency transport, or acting as an additional scout in areas that are unsafe for humans.

In order for the ASM to work within a team setting, the project seeks to provide robots several important capabilities to promote teamwork with humans. One of these capabilities is the ability to anticipate the location of each squad member and make judgements as to the state of a mission. This research aims to give the ASM the ability to judge and reason about the progress of the mission in order to improve its decision-making capabilities.

For example, in one scenario, the ASM robot is patrolling along with its squad members on the kilometer-by-kilometer terrain represented in Figure 1. Location on the terrain is represented in a Cartesian coordinate plane. Suppose the squad, beginning at $(800,400)$, has to visit a sequence of two waypoints, waypoint 1 at $(600,600)$, and waypoint 2, or the terminal waypoint, at $(400,600)$. Suppose that, before or during a mission, the ASM would like to assess the progress of the mission—e.g. "*if I'm in location x, what is my expected time to arrival at Waypoint 1?*" or, a more general question—e.g. "*are we falling behind or moving ahead of the estimated schedule?*" A straight-line approximation of the squad members with respect to the intended location may provide an approximate measure of the rate and overall progress; however, this approximation will not take into account the traversability of the terrain ahead.

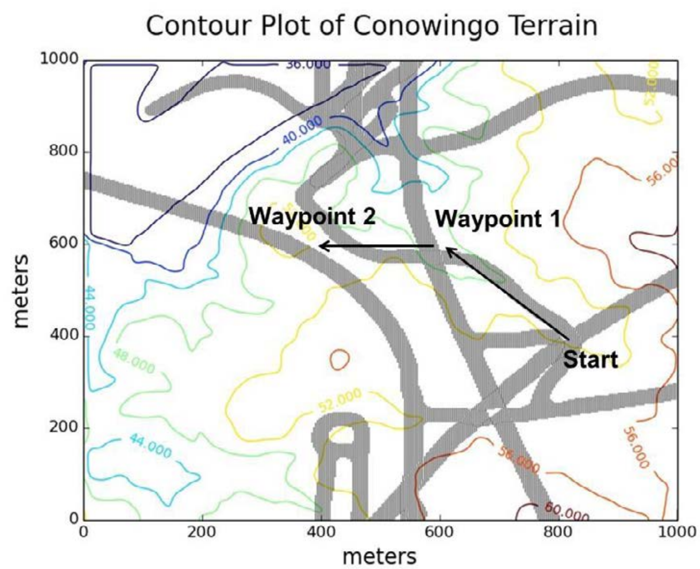


Figure 1: Map of the Conowingo Dam in Maryland. Contour lines of the same color are of equal elevation. Grey areas indicate roads. In this particular patrolling mission, the squad is instructed to first go to Waypoint 1 then to Waypoint 2.

The main goal of this research is to develop a method to reason about the time-dependent location of the squad members with respect to the terrain and mission, allowing the ASM to answer the questions posed previously. It provides a method to predict the set of likely positions of the unit throughout the mission, and the arrival times for each of those positions, so as to be able to anticipate the locations of the squad members. Thus, this paper makes the following contributions:

1. **Create a generalized movement model based on three movement factors.** Section 2.1 introduces a three-factor (mission, terrain, human factors) model of squad member movement. These three factors interact to determine the path of the squad members. Perturbations of these factors may result in the variability of the path of the squad members.
2. **Develop a modified A* algorithm to determine expected paths and location.** Section 2.2 briefly explains the A^* search algorithm—a popular graph traversal algorithm to find the optimal path. This study leverages the A^* process to find the expected location of the squad members. Section 2.2.1 describes a method to construct a portfolio of paths by way of varying the weights of the factors determining movement, as described in Section 2.1. The method then defines a boundary around the area where squad members are likely to operate.
3. **Develop a probabilistic tracking of time.** To further bind the scope of the mission, Section 2.3 develops a method to estimate time to arrive at certain nodes. A simulation of the time of arrivals at varying speed creates a probability distribution providing a range of area where the squad members are operating at a given time.

Successful completion of this body of work will result in a methodology that will enable the ASM the ability to reason about the time-dependent location, and progress of the overall mission.

2. Methods

This section explains the method used to reason about the movement of the squad. First, we explain the three factors in our model affecting movement. The method involves multiple iterations of the A* algorithm—a common cost-based path planning. Varying the cost functions representing unique traversal tactics (i.e. “stay on roads”, “stay in treeline”, or “avoid slopes”) would provide a portfolio of simulated, diverse paths a squad may take to accomplish a mission. The method also explains how we leverage the A* procedure to generate likely time of arrivals for each node.

2.1 Movement Factors

The ASM is required to have some ability to reason about the squad’s objectives, the activities of its squad members, and the environmental context so that the ASM can move in a manner that supports the mission. This reasoning will be based on mission, terrain, and human factors.

Mission Factors are designated and implied actions the unit must take to complete its mission. Designated actions include movement to unit control points such as checkpoints, objectives, and named areas of interest (NAIs). As part of a permission briefing, the ASM is provided the location and actions associated with each of these points. Implied actions are actions depending on the mission context: move quickly, avoid being seen, stay on or off road, etc.

Terrain Factors are constraints and costs imposed by the. The ASM will begin with a map of the terrain, updating the map as the mission progresses based on input from its perception system. Using the map, the ASM will estimate the cost of movement based on distance to the goal, type of soil, land cover, slope, visibility, and other terrain knowledge which might present a benefit or cost with respect to unit movement.

Human Factors represent the unit specific aspects of the model. This includes speed of movement over various types of terrain, restrictions imposed by members with limited mobility, effects of fatigue, and other unit information. Assuming that human factors will be learned by training with the unit, we simplify the impact of these factors by only considering their effect on unit speed.

Mission, terrain, and human factors are used to create a model of unit movement consisting of possible paths and a distribution of times of arrival for areas in a map (see Figure 2).

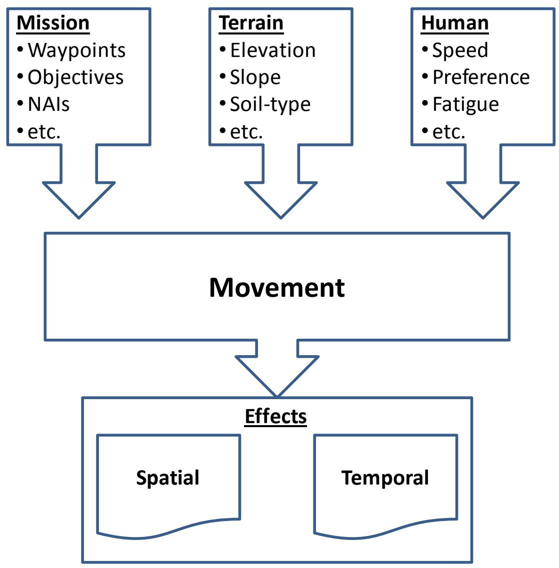


Figure 2: Top: Factors affecting movement; Bottom: Observable effects of unit movement

Our method treats mission control points as planning waypoints, and represents terrain factors as costs of traversing the represented piece of terrain. The relative importance of these costs is determined by the mission contexts, and is represented as weights on various costs. Given a set of weights and costs, the A^* planning algorithm generates a path respecting the unit's mission context. Varying the set of weights and costs produces a population of feasible paths that can be used to predict the unit's future position until it reaches the goal. The method predicts time of arrival at each location based on the human factors model, and retains information about which cells were chosen according to which set of weights, so that the actual path and travel time of the unit can be used to update the ASM's assumptions about mission context and human factors.

2.2 A* Search Algorithm

For input to our algorithm, we subdivide a map into a matrix of cells, each cell covering an equally sized area of land. Each cell represents a node in a graph, connected to the cells adjacent to it vertically, horizontally, and diagonally. Given such a graph, the A^* algorithm can find a least cost path from the start node to the ending node. Given a start point, end point, and a cost matrix, A^* evaluates adjacent cells based on a cost function whose arguments are the present cell (x, y) and the cost of traversing that cell $v := v(x, y)$. This cost function $F(x, y, v)$ can be decomposed into two parts: a backward looking function which accurately tracks the minimal cost of reaching a given cell from the start cell (x_0, y_0), and a forward looking heuristic which estimates the cost of getting from the present node to the terminal node (x_T, y_T). The heuristic function is constrained by the requirement that it never overestimates the actual cost of travel. Denote the backward looking component of F by $G(x, y, v, x_0, y_0)$, and the heuristic component by $H(x, y, x_T, y_T)$ so that

$$F(x, y, v) := F(x, y, v, x_0, y_0, x_T, y_T) = G(x, y, v, x_0, y_0) + H(x, y, x_T, y_T).$$

Beginning with the start node, A^* opens all adjacent nodes for exploration, adding them to a set of nodes known as the $open\ set$. It evaluates the cost of moving to each of these nodes (using F), and closes the lowest cost node, adding it the *closed list*, and opening all cells adjacent to it. The starting node constitutes the root of a tree that will be grown through the search. As a node is added to the closed list, it is designated as a parent node, and the cells opened around it as its children nodes. This process begins with the start node being added to the $closed\ list$ at the beginning, and ends when the terminal node is added to the $closed\ list$. The result is a tree with one path from the root (starting node) to each leaf, of which the terminal node is one. This path constitutes a least cost path through the graph.

Finally, since the central concern is in identifying the cells which might reasonably be entered during the course of the mission, define *mission locale* as the set of all cells opened during any one of the iterations of the A^* process. Our assumption is that given correct weights and cost functions, this locale is the area that the unit is likely to traverse.

2.2.1 Modified A search algorithm*

To produce a correct model of unit movement, the A^* algorithm requires costs of movement and an appropriate heuristic for estimating those costs. As there are many possible factors a unit might consider when planning movement, the method will employ A^* with a range of different cost functions. The model considers: 1) distance, 2) ground slope, and 3) terrain type. At this stage of the development of the model, the model carries simple estimates of these costs which are given in detail in Table 1. The effect of each of the factors on travel time will be discussed in Section 2.3.

Table 1: Summary of terrain attributes affecting cost and time of arrival

Variable	*Description*	*Value*
Distance	The distance between two adjacent nodes	$v_1(\geq 1)$ distance between nodes in meters
Slope	The maximum angle between adjacent nodes.	$v_2 = 1 + \frac{\lvert slope-10\rvert}{90}$
Type	The classification of a node's soil type	$v_3 = \{1(= road),\ 2(= non-road)\}$

Applying the cost definitions of Table 1 to each cell gives us an array consisting of three matrices, each having the same dimensions as the underlying map. Denoting this array of costs by C, the cost of traversing map cell (x, y), with respect to the m^{th} cost matrix (M), is $v_m := C(x, y, m)$:

$$F_m(x, y, v_m) = G_m(x, y, v_m, x_0, y_0) + H_m(x, y, x_T, y_T).$$

Letting d represent the costs of distance, the straight line distance between the current node and the goal node is an admissible heuristic function, or a heuristic function that allows the A* process to find the optimal path, which we denote by $H_d(x, y, x_T, y_T)$. Setting all cell costs $v_i > 1$ makes this distance heuristic an admissible heuristic for all costs M (see Norvig and Russell 2010 for details on admissible heuristics). With this modification, $F_m(x, y, v_m)$ must be changed to

$$F_d(x, y, v_m) = G_m(x, y, v_m, x_0, y_0) + H_d(x, y, x_T, y_T)$$

This cost function can be used to find a path using any single one of the cost matrices, and different cost can be combined by defining a cost function that is a convex combination of the costs in $F_d(x, y, v_m)$. Let $\lambda = (\lambda_1, \dots, \lambda_k)$ (summing to 1) represent the weights for k costs ($k = 4$ in our case). Using a convex combination of cost functions, we obtain the cost function below:

$$F_\lambda(x, y, \boldsymbol{v}) = \sum_{m=1}^{k} \lambda_m \, G_m(x, y, \boldsymbol{v}_m, x_0, y_0) + \ H_d(x, y, x_T, y_T)$$

where v denotes the vector of costs $(v_1, \dots, v_k)$ for the cell (x, y), and the sum is taken over the different cost matrices M.

Choosing a variety of combinations λ produces a population of paths corresponding to different possible preferences of the unit leader. This population provides bounds on the space over which the unit is expected to travel. Since the closed list provides us a natural bound for the portfolio of paths, we define the set of nodes in the closed list as the boundary for the mission locale.

2.3 Time Tracking

In order to form a joint estimate of the unit's position over time as well as space, the distribution of the time of arrival is tracked for each node within the mission locale (defined in Section 2.2).

The mission, terrain, and human factors mentioned in Section 2.1 affect travel time in two ways. First, they determine the order in which cells are traversed—i.e. the order in which the cells appear in the path chosen by A^*. Second, these factors determine the speed with which a unit will traverse cells along a given path. Thus, this study develops a simple model based on travel across one cell, and applies this model over various paths.

For movement across a given cell, we define the base speed s as a parameter representing the speed of unit movement with human factors taken into account but without consideration of terrain effects.

The base speed is modified by the various costs $\boldsymbol{v}$. Represent travel time across a cell, t, as

$$t(x, y, \boldsymbol{v}, s) := \frac{1}{s} \, \ell(x, y, \boldsymbol{v})$$

where ℓ is a function of costs of traversal across the cell. The value of the parameter s may vary from cell to cell, and in the absence of knowledge of these factors, we could treat it as a random variable which is independent and identically distributed for all cells. The influence of the mission context does not appear in $t(x, y, \boldsymbol{v})$ but is incorporated into the values of λ. The weights' influence comes in the choice of paths, by determining which cells are visited. The time to finish traveling across a cell (x, y), whose parent cell was (x_p, y_p), is

$$T(x, y, \boldsymbol{v}, s) := t(x, y, v, s) + T(x_p, y_p, v, s)$$

where the function T on the right side is defined as 0 when computing travel time for the root node (x_0, y_0).

Given the cost matrices and path $(x_0, y_0), \ldots\ (x_n, y_n)$, $T(x, y, \boldsymbol{v}, s)$ can also be written as:

$$T(x, y, \boldsymbol{v}, s) = \sum_{i=0}^{n} \frac{1}{s_i}\ \ell(x_i, y_i, v).$$

$$s_i \sim i.i.d.\ , U(s_{min,} s_{max})$$

In order to determine a distribution of arrival times at a given cell, one could conduct a Monte Carlo simulation with randomly chosen values of s and λ. In Section 3, we treat the selection of s and λ values as a problem in experimental design, in which we vary these parameters in order to cover the space of possible mission paths and location arrival times. Our choice was also influenced by our need to perform minimal computation online during the mission, which limits the amount of computational time available.

After the simulation, each node in the closed list would carry a distribution of times of arrival. The distribution of time of arrival is modeled as a Gamma Distribution. The two-parameter, (k, θ), Gamma distribution are estimated using the approximation for each t_i time of arrivals for each node (Minka 2002):

$$\hat{\theta} = \frac{1}{kN} \sum_{i=1}^{N} x_i$$

$$k \approx \frac{3 - s + \sqrt{(s-3)^2 + 24s}}{12s}$$

where

$$\omega = \ln\left(\frac{1}{N}\sum_{i=1}^{N} x_i\right) - \frac{1}{N}\sum_{i=1}^{N} \ln(x_i)$$

We use this probability to reason about the time of arrivals for each node.

3. Results

This section applies the method described is Section 2 to a real-world terrain. Figure 1 shows the topology of the terrain for the mission, which is an approximately 1 km area of land near the Conowingo Dam in Maryland. The terrain data includes elevation, summarized by the contour lines, and denotes roads by grey paths. The unit's mission is to go from the initial position $(800, 400)$, to waypoint 1 $(600, 600)$, and to waypoint 2 $(400, 600)$. The terrain and human factors determining the choice of path are distance, use of roads, and slope. Table 1 shows the contribution of these variables toward cost, and here we describe our model for their effect on travel time.

In this example, the represented effects of terrain on time, ℓ, as

$$\ell(\boldsymbol{v}, x, y) = \left(v_1 + \left(\frac{slope - \alpha_1}{\alpha_2} \right)^2 + v_3 \right)$$

where v_1 and v_3 represent the value of the distance and type cost matrices at the cell (x, y), as described in Table 1. The contribution of distance is constant, $v_1 = 1$ for the distance of one cell. The contribution of slope is the quadratic term, minimized when the slope is $\alpha_1 = -10.0$, and scaled by $\alpha_2 = 5.0$ so as to decrease the penalty per degree of slope. Finally, v_3 is 1 for travel on a road, and 2 for travel off-road. We have modeled the effects of visibility as a determinant in the choice of path, but not as an influence on the travel time given a path. We let each λ_i assume five values $\{0, .25, .5, .75, 1\}$, such that $\lambda_1 + \cdots + \lambda_4 = 1$ with a Scheffe $\{4,4\}$ simplex consisting of 35 sample points (Wheeler 2014). For each point in the simplex, we generated an A^* path with the appropriate combination of λ values. On each path, we calculated travel time to each cell for base speed $s \in \{.5,1,1.5,2,2.5,3,3.5\}$. Since s is constant, time to arrival, T, is

$$T(x, y, \boldsymbol{v}, s) = \frac{1}{s} \sum_{i=1}^{n} \ell(x_i, y_i, \boldsymbol{v})$$

which can be calculated once for each cell on the path, then linearly scaled to get estimates for other values of s. For a given cell within the mission locale, there is a distribution of possible arrival times.

Using this method, we attain the boundary in Figure 3, which shows the generated mission locale, consisting of the cells within the closed list—ensuring that all optimal paths are within the boundary. Each node in the mission locale carries a Gamma distributed time of arrival. The ASM may now have the elements to execute spatio-temporal reasoning. For example, carrying through our mission example: The ASM may assess that the probability that the squad members arrive at node $n = (700, 500)$ by time $t = \{200,250,300,350,400\}$ resulting in the cumulative probability found in Table 2.

Table 2: Probabilities of time of arrival for n=(700, 500)

t	*F(x|t)*
200	0.00
250	0.05
300	0.45
350	0.89
400	0.99

The ASM may interpret that the probability that the squad members arriving at $t \leq 200$ as low. Appendix A depicts selected examples of bounding expected location using a confidence interval.

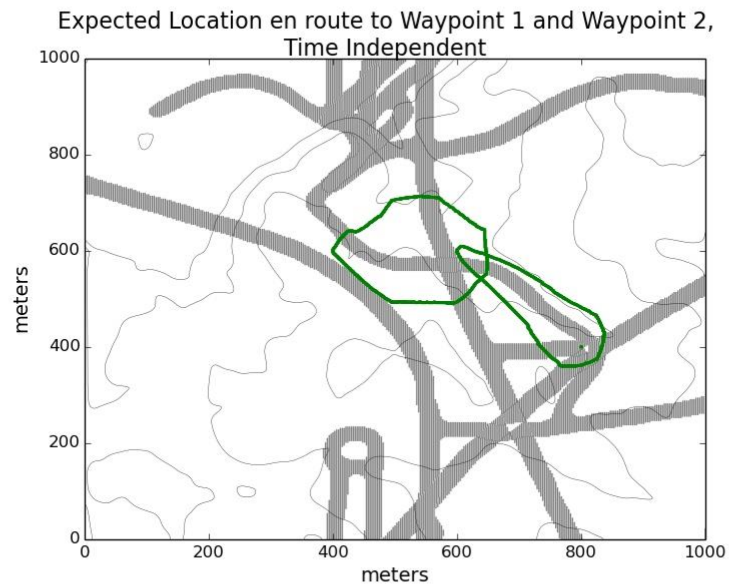


Figure 3: Expected Location of the squad members given mission, terrain, and human. The red boundary depicts the time-independent location of the squad members from the initial position to waypoint 1 and subsequently to waypoint 2.

4. Conclusion and Future Work

This study examined the interaction of the three main factors affecting a mission behavior. The method examined here requires the application of the A* search algorithm to find likely paths, from which we establish a mission locale in which the squad member is likely to operate, while also tracking the distribution of arrival times for each node.

The main motivation behind developing the predictive movement model is to develop tools to begin allowing the ASM to reason about the troop location with respect to various mission, terrain, and human factors.

Future work will involve the improvement of the time tracking model to take into account a better measure of how terrain and human factors affect speed of the soldiers and the augmentation of the variables in the movement factors. Both should improve the accuracy of the location and time estimation of the troops.

Acknowledgements

This project is generously funded by ORAU, Office of Secretary of Defense, and the Army Research Lab.

References

Barnes, M., Florian, J., Chen, J. Y. C., Haas, E., Cosenzo, K. 2010. Soldier-robot teams: Six years of research. In *Proceedings of Human Factors and Ergonomics Society Annual Meeting*, vol. 54, no. 19. 1493-1497.

Childers, M., Bodt, B., Camden, R. 2011. Assessing unmanned ground vehicle tactical behaviors performance. In *Proceedings of the 10th Performance Metrics for Intelligent Systems Workshop*. vol 16. no. 2, 52-66.

Fields, M. A., Haas, E., Hill, S., Stachowiak, C., Laura, B. 2009. Effective robot team control methodologies for battlefield applications. 5862 – 5867.

Minka, T. 2002. Estimating a Gamma Distribution. http://research.microsoft.com/en-us/um/people/minka/papers/minka-gamma.pdf.

Parasuraman, R., Barnes, M., Cosenzo, K. 2002. Adaptive automation for human-robot teaming in future command and control systems. The International C2 Journal. vol. 1. no. 2.

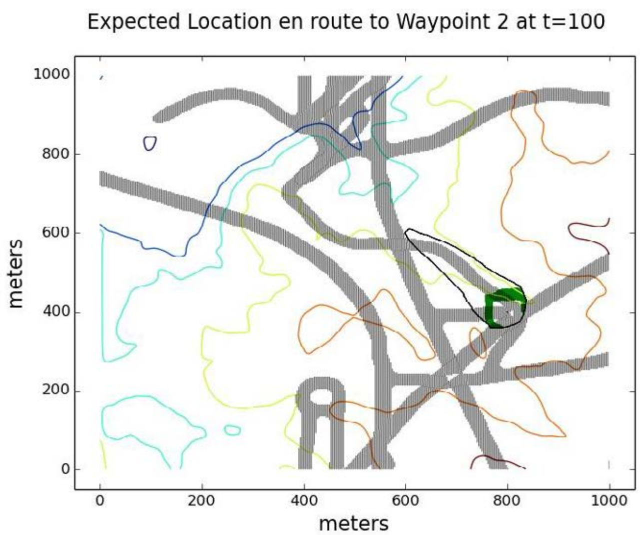
Russel, S. J., Norvig, P. 2010. Artificial Intelligence: A Modern Approach. Prentice Hall 3rd edition.

Wheeler, B. CRAN – package AlgDesign, 2014.

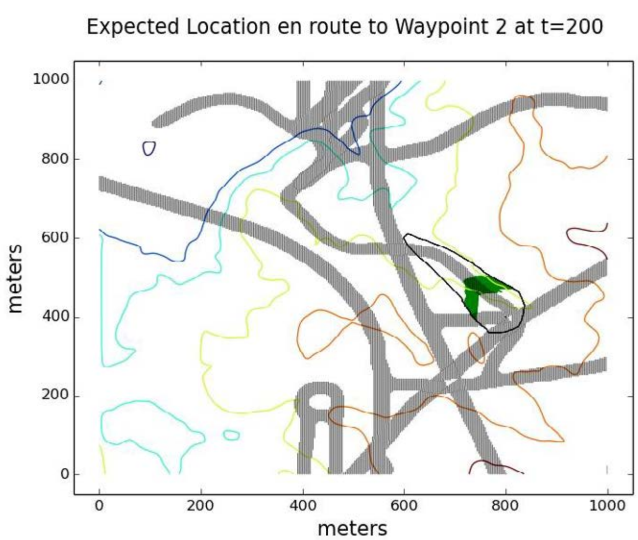
Appendix

The figures below show the two-tailed confidence interval for the probabilistic location of the small unit. Figure 4(a)-(e) depicts the probabilistic location of the small unit en route from the initial position to Waypoint 1 during 100, 200, 300, 400, and 500 time units. The two-tailed confidence interval of $\alpha = .1$. When Waypoint 1 is complete, Figure 5(a)-(e) depicts the probabilistic location of the small unit en route from Waypoint 1 to Waypoint 2 during 100, 200, 300, 400, and 500 time units after arriving at Waypoint 1, again using the two-tailed confidence interval of $\alpha = .1$.

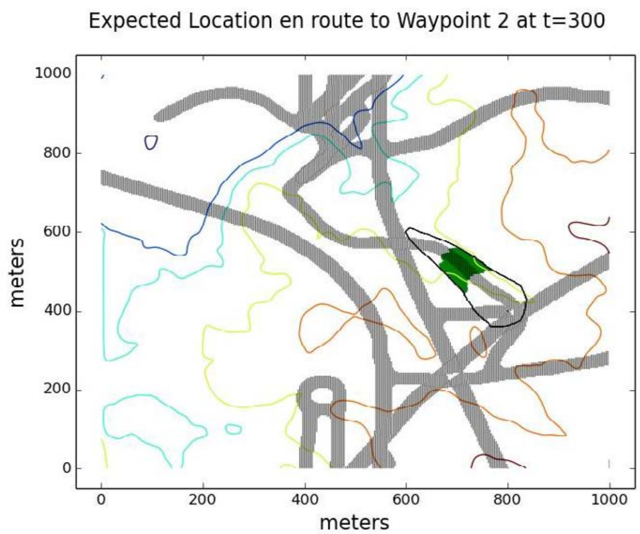
Figure 4(a)-(e): Estimated location en route to waypoint 1 for $t = \{100,200,300,400,500\}$ with $\alpha = .1$



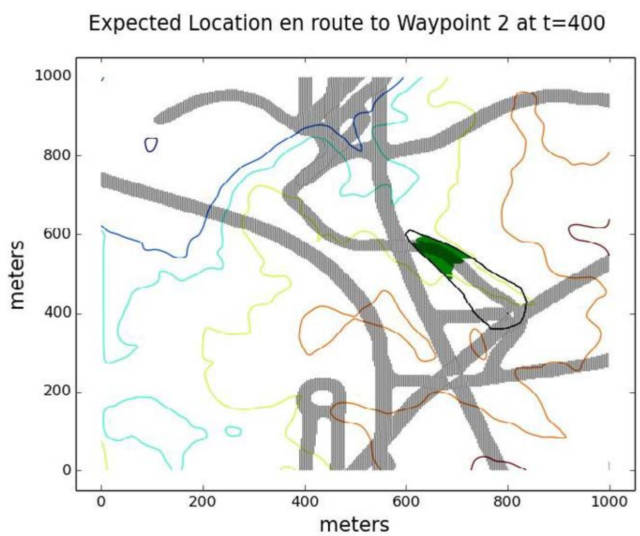
(a)



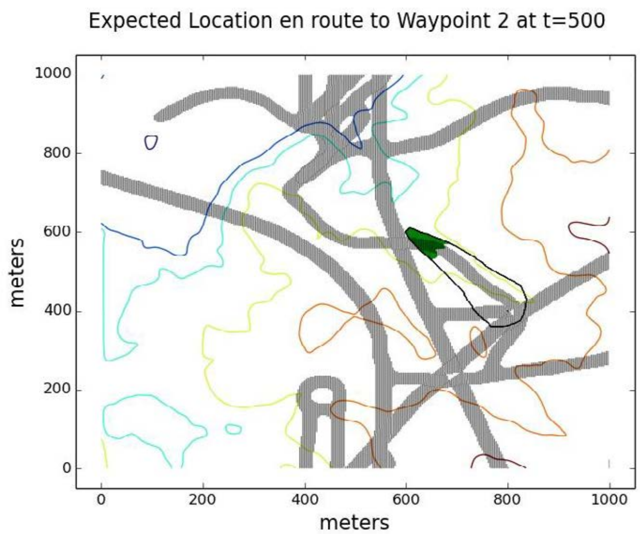
(b)


Expected Location en route to Waypoint 2 at t=300
meters
meters
0
200
400
600
800
1000

(c)

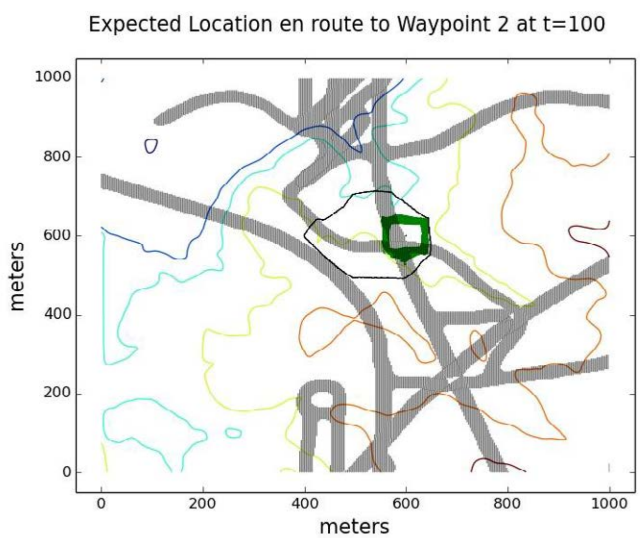

Expected Location en route to Waypoint 2 at t=400
meters
meters
0
200
400
600
800
1000

(d)

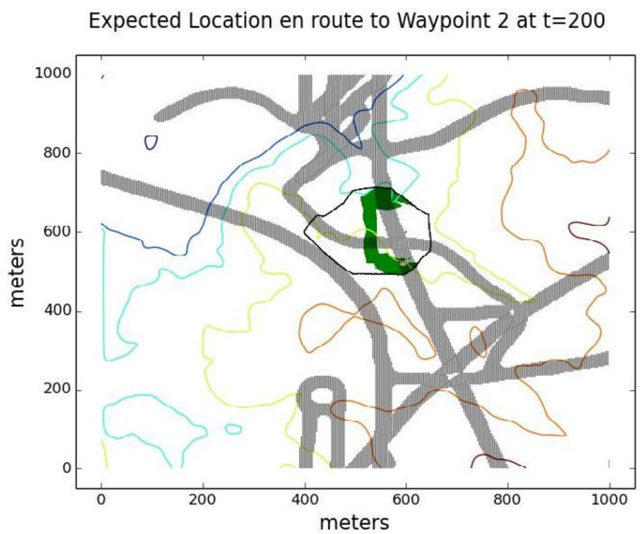


(e)

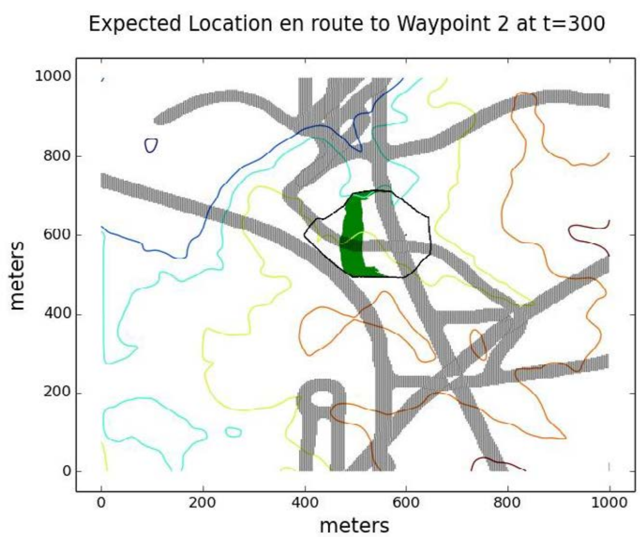
Figure 5(a)-(e): Estimated location en route to waypoint 2 for $t = \{100,200,300,400,500\}$ with $\alpha = .1$



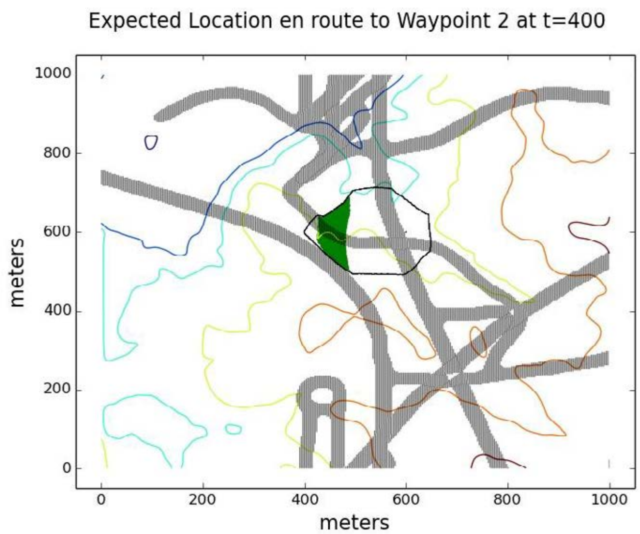
(a)


Expected Location en route to Waypoint 2 at t=200
meters
meters
0
200
400
600
800
1000

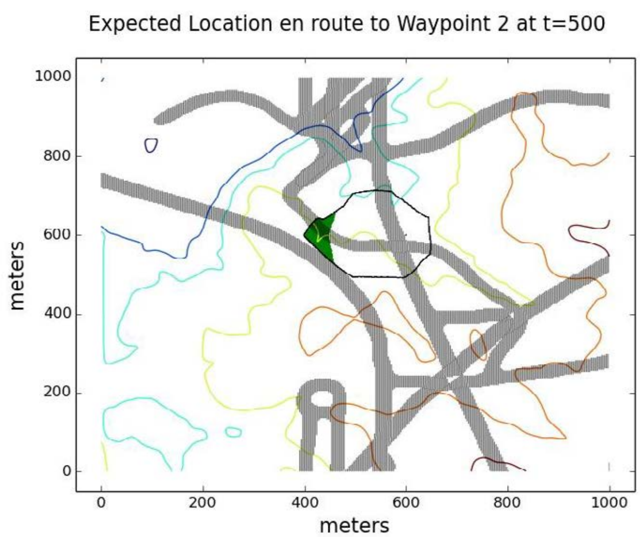
(b)


Expected Location en route to Waypoint 2 at t=300
meters
meters
0
200
400
600
800
1000

(c)


Expected Location en route to Waypoint 2 at t=400
meters
meters
0
200
400
600
800
1000
0
200
400
600
800
1000

(d)


Expected Location en route to Waypoint 2 at t=500
meters
meters
0
200
400
600
800
1000
0
200
400
600
800
1000

(e)