IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 30, 2016, accepted January 20, 2017, date of publication January 25, 2017, date of current version March 13, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2656999

A Green Ant-Based method for Path Planning of Unmanned Ground Vehicles

MOHAMMAD RZEA JABBARPOUR[1], HOUMAN ZARRABI[1], JASON J. JUNG[2], AND PANKOO KIM[3]

[1]Iran Telecommunication Research Center, Tehran 14155-3961, Iran
[2]Department of Computer Engineering, Chung-Ang University, Seoul 06974, South Korea
[3]Department of Computer Engineering, Chosun University, Gwangju 61452, South Korea

Corresponding author: P. Kim (pkkim@chosun.ac.kr)

This work was supported by the research fund from Chosun University in 2015.

ABSTRACT Planning of optimal/shortest path is required for proper operation of unmanned ground vehicles (UGVs). Although most of the existing approaches provide proper path planning strategy, they cannot guarantee reduction of consumed energy by UGVs, which is provided via onboard battery with constraint power. Hence, in this paper, a new ant-based path planning approach that considers UGV energy consumption in its planning strategy is proposed. This method is called Green Ant (G-Ant) and integrates an ant-based algorithm with a power/energy consumption prediction model to reach its main goal, which is providing a collision-free shortest path with low power consumption. G-Ant is evaluated and validated via simulation tools. Its performance is compared with ant colony optimization, genetic algorithm, and particle swarm optimization approaches. Various scenarios were simulated to evaluate G-Ant performance in terms of UGV travel time, travel length, computational time by taking into account different numbers of iterations, different numbers of obstacle, and different population sizes. The obtained results show that the G-Ant outperforms the existing methods in terms of travel length and number of iteration.

INDEX TERMS Path planning, intelligent vehicles, evolutionary computation, ant colony optimization.

I. INTRODUCTION

Recent improvements in hardware, software, sensors and communication technologies lead to many improvements and developments in vehicles technologies. Autonomous vehicles are one of these technologies that attracted many researchers and developers in recent years and combine different fields including computer science, robotics and electrical and mechanical engineering. These vehicles can be count as an evolution in existing vehicle technology and advanced driver assistant systems, and can be used for various purposes and applications ranging from real-time data collection, entertainment, defense, military and delivery [1]–[3]. Autonomous or unmanned vehicles can be divided into three main categories, namely, Unmanned Underwater Vehicles (UUVs), Unmanned Ground Vehicles (UGVs) and Unmanned Aerial Vehicles (UAVs). UUVs are utilized for ocean exploration, under water construction, oceanography and military applications for long time [4]. These days, UGVs are being studied by the DARPA (Defense Advanced Research Projects Agency), the US military, for search and rescue operations, cargo and packet delivery, bomb detection and other military applications [5]. UAVs similar to UGVs can be used in data collection and military applications. In addition, UAVs are being utilized for remote sensing, imagery collection, map creation and also for data transmission as mobile sinks [6].

Among all types of unmanned vehicles, our focus in this paper is on UGVs. UGV navigation is one of the main problems in this research field since navigation is an essential step in the most of the existing UGVs applications. Generally, UGV navigation includes the procedure of consecutive motion that guides the UGV from origin point to destination point through collision-free path in configuration area which contains obstacles. These obstacles can be static or dynamic. According to the study in [7], UGV navigation problem can be divided into three sub-problems: 1) World perception: in this stage, UGV senses the surrounding environment in order to identify the existing obstacles and paths, 2) Path planning & generation: the gathered data in the previous stage is utilized to create an ordered sequence of intermediate points that the UGV must visit and reach them to generate a collision-free path from origin to destination, 3) Motion controlling: this stage controls the UGV actions and movements to make sure that UGV follows the correct path.

The main concentration of this paper in on the second subproblem of UGV navigation, namely, path planning and generation. UGV path planning is classified as follows: global and local path planning. In the former, prior knowledge regarding the configuration area and predefined collision-free paths are provided for UGV. In contrast, in the latter case, configuration area is partially or totally unknown and the path planning scheme uses sensory extracted information to find a collision-free path for UGVs. Various approaches have been proposed and developed in the literature in order to solve this problem in both static and dynamic environments. These solutions can be divided into four categories: visibility graphs, potential field, cell decomposition and heuristic approaches [8]. These approaches are completely discussed and criticized in next Section. Although most of the existing approaches in the literature obtained promising results for collision-free path planning, they suffer from high computational cost, trapping into local optimum/minimum condition, and most of them are suitable for 2D environments and maps [9]. In addition, these approaches do not pay attention to one of the main factors of UGVs which is limited on-board energy/power. Hence, finding a proper solution that considers power consumption in its path planning is still lacking.

In this paper, a new approach for solving path planning problem of UGVs is proposed. In this approach, Ant algorithm is integrated with power consumption prediction model, called G-Ant (Green Ant), for considering power consumption and green environment issues in path planning procedure of UGVs. The main goal of G-Ant includes proposing the collision-free shortest path with low travel time and power consumption as well as smoother travel speed. Based on the existing papers in the literature, there was not any similar approach that exploit power consumption in its path planning procedure which is the main contribution of this paper. The effectiveness of the proposed approach is demonstrated via simulation environment and its validity is examined by comparing it with other intelligent algorithms.

The reminder of the paper is organized as follows: Section 2 discusses the existing related works. An overview of ant algorithm along with power consumption prediction model is provided in Section 3. Our proposed approach, G-Ant, is discussed in Section 4. Section 5 contains the simulation results and algorithm validation. Finally, Section 6 concludes the paper and provides the future direction of this research.

mds
August 26, 2015

II. PATH PLANNING APPROACHES

Existing path planning approaches can be classified into four types: visibility graphs, potential field, cell decomposition and heuristic approaches [8]. In first case, a set of lines is defined in free space in a way that it connects an object's features to those of another [10]. The complexity of these approaches are high since $O(n^2)$ links are created for n features. Potential functions are utilized by potential field approaches. These functions are suitable for obstacle avoidance in static configuration areas. UGVs or robots are considered as point in a potential field which integrates tension to the destination, and expulsion from obstacles. For example, semilunar and normal functions are used in [11] to form the vector fields which control the speed and heading direction of UGVs. In [12], the distance to the destination along with the obstacles' angular width are utilized to calculate a potential field for UGV heading which controls angular acceleration, direction towards the destination and avoiding from obstacles. Although potential functions provide proper solutions, high order potentials are required to obtain high accuracy which increases the computational complexity and also UGVs are sometimes trapped in local minimum condition before finding a proper path to the destination. Hence, heuristic approaches are developed to overcome the mentioned drawback of potential functions. The configuration area is represented by occupancy grid which divided into a grid of equally separated cells. In this way, the path planning problem converted to a graph search problem where A* algorithm [13] is most favorable solution. A* is a cornerstone of some other algorithms such as best-first, depth-first and breadth-first that can be used for problem solving. Linear incorporation is used to extend the D* algorithm [14], an extension of A*, during each vertex expansion in [15] as an example of heuristic approaches. High time complexity is the main drawback of heuristic approaches. In the last type, the configuration area is divided into regions and in each region, any contact between UGV and obstacles are identified. Retraction of free space onto the Voronoi diagram, which is constructed through the time evolution of Cellular Automata (CA) [16] is used in [8]. However, they do not pay attention to the drawback of Voronoi diagram in high dimensions which requires too complex data structure.

In the last decade, various artificial intelligence algorithms were utilized to solve UGVs and robots path planning problem. Fuzzy logic system or its combination with other techniques are used in several researches [17]–[20] for guiding mobile robot (e.g. UGV) from origin to destination in either unknown or known environments. However, similar to most of the other approaches, various and several sensors should be attached to UGV in order to surrounding identification which impose high cost to the system. Moreover, selection and dispensation of membership functions and rule organization have a direct impact on fuzzy logic systems performance. Hence, most of the researchers tried to enhance these factors, individually or simultaneously. Approaches based on neural network for path planning are also proposed in various research works [21]–[24] without considering its high computational cost. In addition, although neural network architecture and synaptic weights of the connecting nodes have direct impact on neural based approaches performance, identifying optimal selection is a sophisticated task. Multilayered CA which contains lattice of cells and four layers of identical grid for presentation of configuration area and for solving path planning problem of robots, respectively, is proposed in [25].

Bio-inspired algorithms such as Ant Colony Optimization (ACO) [26]–[31], Genetic algorithm [32], [33], Artificial Immune System [34], [35], Cuckoo Search [36], Bacteria Foraging Optimization [37], and Particle Swarm Optimization [38], [39] are also utilized in UGV and robot path planning research area. A comprehensive survey regarding UGVs/robots path planning can be found in [40]. Since our proposed solution in this paper is based on ACO algorithm, ACO based path planning approach are discussed in more details in the following.

The feasibility of using ant algorithm for UGV/robot path planning in 2D environment is investigated in [26]. High searching speed and local optimization obviation are two main advantages of this approach. The mathematical model as well as ant algorithm have been established and presented. Another ant-based path planning in static environments is proposed in [27] where the information of environment constrains and neural network calculates objective function by considering path length. In addition, the path nodes are considered as an ant, hence with the quality of ant algorithm, the best path is calculated. Visiting multiple targets in the presence of obstacles via UGVs are investigated in [28]. The configuration area and UGVs are modelled in the form of discrete cells and points, respectively. Position of the targets are known for UGVs but the obstacles and their positions are sensed by embedded sensors. It looks like the travelling salesman problem along with obstacles in configuration area. Simple ACO Distance-Memory (SACOdm) [29] is ant-based path planning approach that includes distance of the source and destination and ants memory, which stores the visited nodes, in its path finding procedure. In addition, criterion of a Fuzzy Inference System, tuned by a Simple Tuning Algorithm, effect on path planning procedure. Combination of ant algorithm and artificial potential field are proposed in [30] for path planning where the former is used as global path planning and the latter is utilized as local path planning strategies. In order to prevent artificial potential field from falling into local optimum, the generated pheromone via ants are used as global information for artificial potential field. Based on the simulation results, this integrated approach outperforms genetic algorithm in path planning.

Another hybrid approach which combines CA and ant algorithm for finding collision-free path for UGVs/robots is presented in [9]. This hybrid approach can be utilized in both static and dynamic environments since it does not need any priori information regarding configuration area and reacts to obstacles distribution changes. CA path planner is utilized when there is not any pheromone on the links of the configuration area. Being autonomous and low complexity are two main advantages of this approach. Another ant-based path planning method that uses a precise representation of heuristic and visibility equation of state transition rules are discussed in [31]. This approach is proposed for static environments and its performance is evaluated in terms of number of iteration and computation time. Predefined number of scout ants, m, collaboratively investigate the configuration area to find a collision-free path for UGVs navigation [41]. Scout ants are divided into two types. n ants use nearest-neighbor search strategy, while, q ants (q=m-n) use random search strategy. These strategies are enhanced by constructing a global taboo list of visited cells. Hence, this method requires a reliable communication among ant agents as well as a central server for global taboo list construction which are difficult in real world scenarios.

III. ANT ALGORITHM AND ENERGY CONSUMPTION MODEL

A. ANT ALGORITHM THEORY

In this section, the theory of ant algorithm along with its procedure are discussed. The way of finding food sources and accumulate them in the nest through the shortest path via real ants have attracted the attention of many researchers and scientists. Based on the existing experiments, real ants deposit a chemical liquid, named pheromone, on their traversed route between the nest and food source according to the found food source quality. In other words, ants communicate and collaborate with each other through sniffing the pheromone trails. The pheromone intensity reduces over the time, called pheromone evaporation, to raise the chance of finding new routes instead of insisting on the found path. This issue that the ants find and use the shortest path between their nest and the food source has been proved both mathematically and experimentally by researchers. Mathematical proof can be found in [42], while double bridge experiment is used for experimental proof. Double bridge experiment is depicted in Figure 1 in which there are two different bridge (i.e. paths) between source (i.e. nest and destination (i.e. food source).

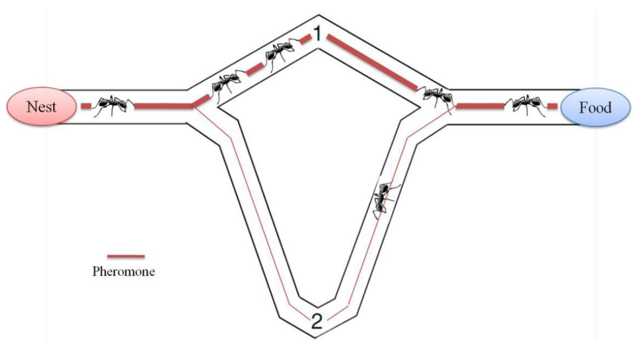


FIGURE 1. Double bridge experiment [43].

Length is the main difference between these two bridges where bridge 1 is shorter than bridge 2. Initially, the ants explore the surrounding environment for finding food sources by performing a random selection between bridges 1 or 2. However, the ants which select bridge 1 arrive faster to the food source and come back to the nest earlier that the ants which choose bridge 2 for food exploration. This is because bridge 1 is shorter than bridge 2. In this way, more ants will be attracted to bridge 1 due to pheromone existence on it. Therefore, the pheromone intensity will be increased on this bridge over the time, while, it will be reduced on bridge 2 due to pheromone evaporation. As a result, ant-based algorithm is an efficient way to find shortest path between two desired points. The behavior of real ants is simulated in [44] and various ant-based algorithms are proposed by researchers such as Ant System (AS), rank-based AS, and Ant Colony Optimization algorithms. The pheromone update procedure and some additional details in the pheromone trails management are the main differences among various ant-based approaches. More information regarding various types of ant algorithms and their differences can be found in [45].

Ant algorithms contain four main steps. Each of these steps are briefly discussed in the following.

1) Environment representation: the surrounding environment should be converted into a graph with N nodes and L links. The nodes indicate start, intermediate and end points, while, links indicate existing paths and routes between nodes.
2) Initialization: at this step, a predefined number of ants (N_{ant}) are located on start point and a weight is allocated to each existing link. Various values can be assigned as initial weight for links such as distance, random number or a number calculated through a formula. At each specified time intervals (TI) ants are regenerated to explore the problem graph for finding proper solution. The value of these variables (i.e. N_{ant} and TI) should be assigned via experiments or trial and error approaches. The ants use probability function to select the next node in their path from origin to destination. Equation 1 can be used for calculating the selection probability of j as next node from i by ant k.

$$y_{ij}^{k}(t) = \begin{cases} \frac{(\tau_{ij})^{\alpha}(\eta_{ij})^{\beta}}{\sum_{h \notin tabu_k}(\tau_{ih})^{\alpha}(\eta_{ih})^{\beta}} & ifj \notin tabu_k, \\ 0 & otherwise, \end{cases} \tag{1}$$

All above mentioned variables are explained in Table 1.

3) Pheromone update: Ants utilize probability function to select next node and store any visited node in their memory. As soon as an ant finds a food source, it evaluates both quantity and quality of the food source and takes some food and starts return trip. During the return trip, each ant deposits a pheromone on the return path based on the obtained information from food source. In ant-based algorithm this procedure is called pheromone update rule which contains two concepts, namely, pheromone reinforcement and pheromone evaporation, at the same time. In the former case, the pheromone intensity of the links which are traversed by ants are increased, while, in the latter case, the pheromone intensity of the other links is reduced. Pheromone update rule has a direct impact on the exploitation (i.e. enhancing found path) and exploration (i.e. discovering new path) characteristics of ant algorithms. Equation 2 represents pheromone update rule considering these two characteristics.

$$\tau_{ij}^{new} = (1-\rho)\tau_{ij}^{old} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \tag{2}$$

All above mentioned variables are explained in Table 1. The amount of pheromone deposited on the link between 2 points, i and j, by ant k is computed via Equation 3.

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q}{f_k} & \text{if the } k^{th} \text{ ant passed link (i,j),} \\ 0 & otherwise, \end{cases} \tag{3}$$

The mentioned variables in Equation 3 are explained in Table 1.

4) Stopping step: Reaching a predefined number of iteration indicates that the ant-based algorithm is completed, while, reaching a predefined maximum number of nodes before arriving to the destination point leads to ant drop.

TABLE 1. Variables of Ant algorithm.

Variable	Description
α	Importance coefficient of pheromone intensity
β	Importance coefficient of route cost
τ_{ij}	Pheromone intensity on link (i,j)
η_{ij}	Cost of link (i,j)
$tabu_k$	Visited nodes table by ant k
ρ	Pheromone evaporation ratio
m	Number of ants
Q	Constant value
f_k	Route cost obtained by ant k

B. UGV ENERGY CONSUMPTION MODEL

The UGV energy consumption is affected by various factors such as situation of road surface, velocity, internal resistance of vehicle, driving style (e.g. aggressive and stop-and-go operations) and embedded electronic equipment [46], [47]. The effect of these factors on fuel/energy consumption has been extensively investigated for conventional vehicles. For example, based on the results in [48] and [49], high speed and aggressive driving increase both emission and fuel consumption compared with moderate speed and normal driving. Moreover, fuel consumption can be increased by average of 5%-40% due to aggressive driving based on the results in [50] and [51], while it can be enhanced by average of 10%-33% in result of eco-driving style [52]. However, these results are not applicable for UGVs due to their tele-operated nature, delay between observation and action, small size and unique dynamics.

One of the main limitations of the battery-powered UGVs is the available onboard battery power. Vehicle locomotion is the main source of UGV energy consumption. As mentioned

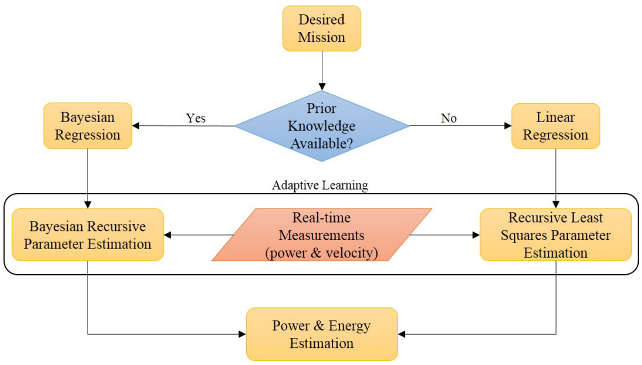


FIGURE 2. **Overview of UGV Energy Consumption Model [53].**

before, the propulsion power of UGVs is affected by various factors. Hence, developing a method for predicting UGV mission energy consumption is a necessity. In this study, the method proposed in [53] is utilized for this purpose. This method is able to predict the UGV energy consumption in the presence and absence of mission prior information including qualitative knowledge regarding the road surface situation and grade, and driving style. Linear regression and Bayesian regression models triggered by longitudinal dynamics are utilized for energy consumption prediction in the absence and presence of mission prior information, respectively. Figure 2 provides an overview of UGV energy consumption model. As it can be seen, the parameters are continuously updated according to real-time data (e.g. velocity and power consumption) in both models in a recursive manner.

TABLE 2. **Variables of equation 4.**

Variable	Description
$P(t)$	Power at time t
$F(t)$	Total traction force
$v(t)$	Vehicle velocity
m	Vehicle mass
$a(t)$	Vehicle acceleration
W	Vehicle weight
$\theta(t)$	Road Grade
f	Road rolling resistance coefficient
C_I	Internak resistance coefficient
b	Consumed energy via onboard equipment
$\varepsilon(t)$	Model error

Road grade, road surface situation, driving style, vehicle internal resistances and embedded sensors and electronic equipment are considered as main UGVs energy consumption factors in [53] and formulized as Equation 4 and its variables are represented in Table 2.

$$\begin{aligned} P(t) &= F(t)v(t) + b \\ &= (Wsin(\theta(t)) + fWcos(\theta(t)) + ma(t) + C_I) \\ &\quad \times v(t) + b + \varepsilon(t) \end{aligned} \tag{4}$$

By considering θ in Equation 4, this equation is nonlinear. This equation can be linearized as Equation 5, since in most cases, this value of this variable does not exceed 15 degrees.

$$\begin{aligned} P(t) &= F(t)v(t) + b \\ &= (W\theta(t) + fW + ma(t) + C_I)v(t) + b + \varepsilon(t) \end{aligned} \tag{5}$$

Equation 5 can be rewritten as a linear regression model as Equation 6.

$$P(t) - ma(t)v(t) = v(t)W(\theta(t) + f + C_I') + b + \varepsilon(t) \tag{6}$$

The left side of above equation can be defined as the response $y(t)$, $v(t)W$ can be defined as the predictor $x(t)$, and $\theta(t) + f + C_I'$ is considered as the regression model donated by C and contains the road grade, rolling resistance coefficient, and internal fractional losses. Thus, Equation 6 can be rewritten as Equation 7.

$$y(t) = b + Cx(t) + \varepsilon(t) \tag{7}$$

In this paper, it is assumed that the prior knowledge is not available, hence, linear regression model (Equation 7) is used for energy consumption prediction. The composed variable, C, and b are updated according to real-time velocity and power values via recursive least squares (RLS) prediction with the forgetting factor, λ_{ff} [54]. The upcoming vehicle velocity is required for energy estimation which can be predicted according to driving style of the UGV operator. Exponentially weighted moving average (EWMA) [55] with a weight λ_u, is utilized for this purpose. More details regarding this issue can be found in [46], [47], and [53]. In most of the cases, the distance between source and destination is known and the position of UGV can be traced by means of GPS. Hence, based on the real-time UGV speed and remaining distance to the destination, the duration of travelling to destination, T_{end}, can be predicted. The total energy consumption, E_c can be computed by integrating the instant power over the duration of the mission as Equation 8.

$$E_c = \int_0^{T_{end}} P(t)dt \approx \sum_{j=1}^{n} P(j)\Delta t \tag{8}$$

In the next section, our new approach for solving path planning problem of UGVs, called G-Ant, is discussed.

IV. GREEN ANT

G-Ant contains four main phases, namely preparation, map exploration, energy consumption calculator, and path planning. Figure 3 illustrates these phases along with their details and relationships. Each of these phases is discussed in the following paragraphs.

1) Preparation: This phase includes two sub-phases, called map provision and data collection. This is the first step of G-Ant approach where the required UGV related data (e.g. velocity, mass, weight and acceleration) for energy consumption calculation are collected and the physical environment is transformed into a configuration space which includes every demonstration of

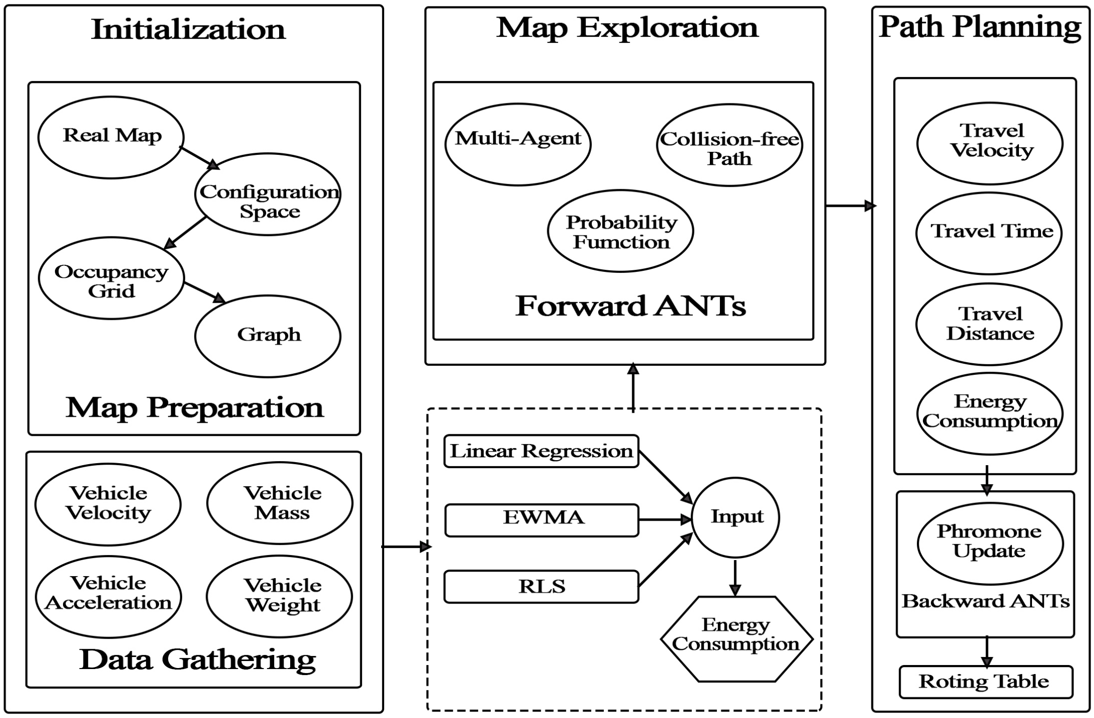


FIGURE 3. G-Ant Framework.

the UGV position, orientation, linear or angular speeds as well as any other measures of interest [56]. For physical map transformation, some of current approaches, such as driving corridors [57], start a search in successive coordinates utilizing the road boundaries and obstacles positions. Other approaches such as Voronoi Diagrams [58], occupancy grids [59], cost maps [60], and lattices [61] which are called decomposition techniques, parse the configuration space with higher resolution. In G-Ant, occupancy grid is utilized for this purpose due to fast discretization and low computational power [62]. In occupancy grid, the configuration space is divided into a grid and each cell is associated with a probability of being occupied by a UGV or an obstacle. Moreover, a point is assigned to each obstacle-free cell and a link is added between two points if and only if their associated cell is adjacent. In this way the configuration space is converted to a graph, $G = (P, L)$, where P and N represent the set of points and links, respectively. An example of this graph is illustrated in Figure 4.

2) Energy Consumption Calculator: The second phase of G-Ant approach is one of our main contributions where UGV energy consumption is calculated and considered in UGV path planning procedure. As mentioned earlier, the energy consumption is estimated via the linear regression model (Equation 7). Since UGV related variables are time dependent, the required time by UGV for traveling from origin to destination is divided into time intervals which are indexed by $k = 1, 2, \ldots$, n to consider the dynamic aspect of these variables. This issue is represented in Figure 5 where R represents remaining distance to destination, Δt and $T_{end} = n\Delta t$ indicate sampling interval and end time of travelling, respectively.

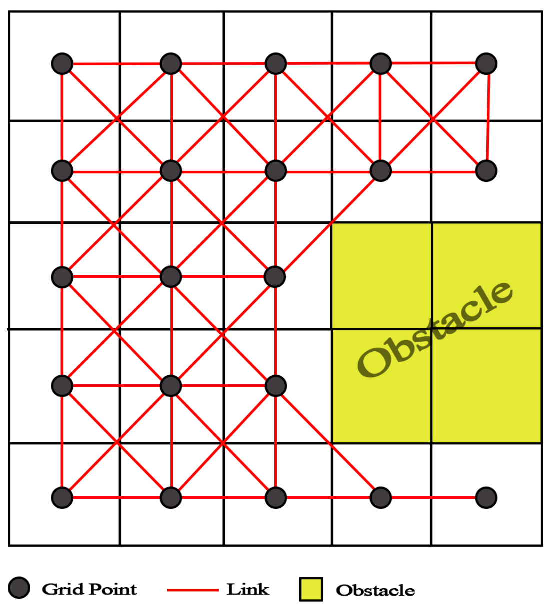


FIGURE 4. Planning graph.

When RSL prediction is utilized, the regression model variables can adjust to small shifts and drifts in the

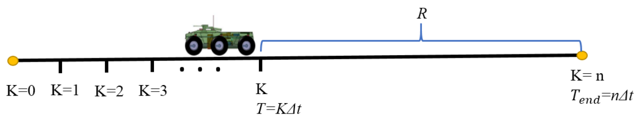


FIGURE 5. UGV travelling from origin to destination considering time intervals.

energy consumption. However, this adjustment can be slow if an unexpected event such as transition from one road segment to another happens. EWMA control chart must be utilized to solve this issue. EWMA controls the estimation residuals based on the study in [63]. The EWMA monitoring statistics, *z(k)*, is computed by Equation 9.

$$z(k) = \lambda_u \hat{e}(k) + (1 - \lambda_u) z(k-1) \tag{9}$$

where,

$$\hat{e}(k) = y(k) - \hat{y}(k)$$
$$\hat{y}(k) = E[y(k)|x(k), \hat{C}(k-1), \hat{b}(k-1)]$$

where λ_u and $\hat{y}(k)$ are the weight of EWMA, and response estimation at k, respectively. $\hat{C}(k-1)$ and $\hat{b}(k-1)$ are the predicted variables at *k-1*. It is worth noting that the RLS covariance matrix is reset to its primitive value if an out-of-control signal is identified. The total energy consumption (Equation 8) can be predicted by Equation 10.

$$\hat{E}_c(k) = E_t(k) + \hat{E}_{rc}(k) \tag{10}$$

where $E_t(k)$ and $\hat{E}_{rc}(k)$ are the consumed energy until time $t = k\Delta t$, and the anticipated required energy to traverse the rest of the path, respectively. By considering Equations 8 and 10, Equations 11 and 12 can be used for estimation of the anticipated overall energy consumption as well as the corresponding variance at k, respectively. More details can be found in [53].

$$\begin{aligned} \hat{E}_c(k) &= (\sum_{j=1}^{k} P(j) + \sum_{j=1}^{\hat{n}-k} \hat{P}(k+j|k))\Delta t \\ &= E_t(k) + (\hat{n} - k)(W\hat{v}(k+1|k)\hat{C}(k) \\ &\quad + \hat{b}(k))\Delta t \end{aligned} \tag{11}$$

$$var(\hat{E}_c(k)) = (\sum_{j=1}^{\hat{n}-k} var(P(k+j|k)))(\Delta t)^2 \tag{12}$$

where $\hat{P}(k + j$â"Ć$k)$ is the j-step-ahead estimation of response and $W\hat{v}(k+1$â"Ć$k)$ is one-step-ahead estimation of predictor. The predicted energy consumption is utilized in next phases.

3) Map Exploration: There are two types of ants in G-Ant, namely Forward ANT (FANT) and Backward ANT (BANT). FANTs are utilized in map exploration phase, while BANTs are used in the next phase, i.e. path planning phase. Based on the gathered data in two previous phases, FANTs are placed on start point and explore the configuration space to find the shortest collision-free green path to the destination by selecting probabilistically the next point to move. The path is green due to considering energy consumption in path finding procedure. FANTs utilize Equation 13 as probability function, PF_{ij}, to select the next point on their path.

$$PF_{ij} = \frac{\alpha(\tau_{ij}) + \beta(\eta_{ij})}{\sum_{h \notin tabu_k} \alpha(\tau_{ih}) + \beta(\eta_{ih})} \times (\frac{N_j}{N_j + 1}) \tag{13}$$

N_j reflects the number of neighbors for next point j and increases the selection probability of the point with more neighbors to become a next point in the path. τ_{ij} is the pheromone intensity on link (i, j) computed by BANT via Equation 15. η_{ij} indicates the instant state of the UGV speed, v_{ij}, and the link length, ll_{ij}. The effect and importance of τ_{ij} and η_{ij} are controlled by weight factors, α and β, respectively. Equation 14 is utilized for ij calculation.

$$\eta_{ij} = \frac{1}{ll_{ij}} + \frac{v_{ij}}{\varphi} \tag{14}$$

Equation 14 represents that probability function has direct relationship with UGV speed, while, it has inverse relationship with travel distance or link length. φ is the maximum speed for UGVs and set to 7.2 *km/h* ($\approx 2m/s$) based on a scaled EPA US06 [46], [47] to normalize this equation.

4) Path Planning: At final phase, when a FAN arrives its destination, it becomes a BANT and comes back to the start point by utilizing its memory. Equation 15 is used via BANTs for updating pheromone intensity of the links. This equation is named pheromone update rule and either increment or reduce the pheromone trail intensities. The pheromone value is increased on the links that traversed by BANTs, called pheromone reinforcement, while the pheromone intensity of the other links is reduced, called pheromone evaporation.

$$\tau_{ij}^{new} = (1 - \rho)\tau_{ij}^{old} + \sum_{k=1}^{n} \Delta\tau_{ij}^{k} \tag{15}$$

ρ is pheromone evaporation coefficient and has constant value between 0 and 1. n indicates the number of points in configuration space. The amount of $\Delta\tau_{ij}^{k}$ is calculated by Equation 16.

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{1}{TT_{ij}^{k}} + \frac{1}{\hat{E}_c(ij)^k} & \text{if the link (i,j) passed by ant k,} \\ 0 & otherwise, \end{cases} \tag{16}$$

TT_{ij}^{k} and $\hat{E}_c(ij)^k$ are travel time and predicted UGV energy consumption. Hence, if a link belongs to a planned path by an ant, its pheromone intensity is updated according to its travel time and UGV energy consumption rate. Path planning table is created based on these obtained data.

The algorithm should be stopped or completed when a predefined situation(s) is reached. Some examples of stopping criteria are as follows: predefined execution time, number of iterations, maximum visited hops and the pheromone value remaining constant for a number of successive iterations. However, G-Ant executes for an infinite number of cycles. In G-Ant each cycle ends by finding a path from origin to destination or reaching a predefined number of visited points or predefined number of iterations. In two last conditions, ant is dropped before reaching its destination. This concept is also useful for algorithm loop prevention.

V. PERFORMANCE EVALUATION

In this section the performance of G-Ant in two and three dimensional configuration space under known environment is evaluated through simulation. MATLAB R2015b processing under Windows 7 is utilized for conducting simulation experiments. G-Ant is implemented on a PC with Intel Core i7 processor running 3.5 GHz, RAM of 8 GB. The experiments are conducted on several environmental scenarios including different number of obstacles, different population size and different number of iterations. The performance of ant based approaches is very sensitive to its various variables. Finding a proper value for these variables may result in faster cost function convergence as well as mitigating local minimum problem. Hence, this stage is one of main tasks in G-Ant. We adopted these values from the study in [64] where various simulation experiments were conducted to investigate the impact of these variables and to find proper value for them. Moreover, Equation 4 is utilized for generating power data of UGVs. Aggressive EPA US06 driving cycle, illustrated in Figure 6, is used for indicating the velocity of UGVs. Moreover, the standard deviation of error term in Equation 4 is near to 10% of the mean consumed power during travelling from origin to destination according to the values from the UGV dynamic model.

The required energy (consumed energy) that UGV needs to travel from origin to destination is estimated and updated via Equations 11 and 12. Table 3 summarizes the variables and their values used in G-Ant to finding the shortest collision-free green path from origin to destination for UGVs. These values are adopted from experimental results in [46] and [64].

The G-Ant performance is validated by comparing it with ant colony optimization (ACO) [27] (i.e. G-Ant without energy consumption model), Genetic Algorithm (GA) [32] and Particle Swarm Optimization (PSO) [27], [38] approaches. These approaches were selected due to their popularity as well as their usage in most of the existing UGV/robot path planning approaches. Various scenarios were simulated to evaluate G-Ant performance in terms of UGV travel time, travel length, computational time by taking

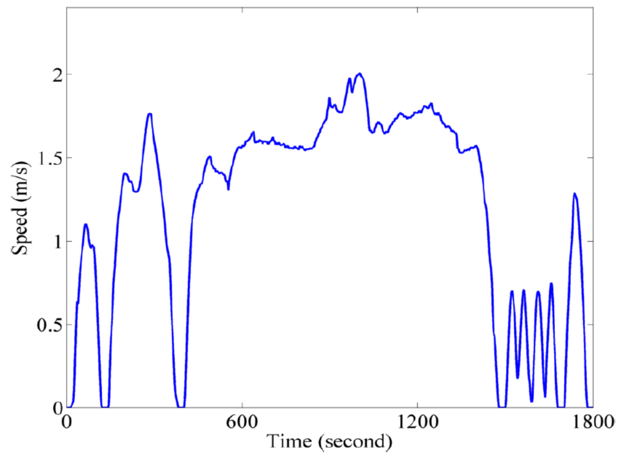


FIGURE 6. Drive cycle based on a scaled EPA US06.

TABLE 3. Configuration parameters and their values in G-Ant.

Variable	Value
α	0.4
β	0.6
ρ	0.3
TI	3 Δt
φ	2 m/s
UGV mass	30 kg
λ_{ff}	0.98
λ_u	0.98
b	0 Watts
Sampling interval (Δt)	10 Second

into account different number of iterations, different number of obstacle, and different population size.

Firstly, the effectiveness of G-Ant has been verified through simulation environment by considering various number of iterations and population size in terms of travel length (distance) and computation time. The configuration space that is used in this stage is represented in Figure 7. Green cells represent start and end points of the path. Their coordination is (0.5,0.5) and (9.5,9.5), respectively. The configuration space size is 10Ã-10 and includes 100 cells, where 16 of them contain obstacle. The orange cells represent obstacles.

Table 4 shows the obtained results for G-Ant considering various number of iterations and population size ranging from 10 to 100 by step 30 (i.e. 10, 40, 70, 100). Since the G-Ant is meta-heuristic approach, every time it is executed it may lead to different trajectory convergence. Hence, the algorithm was executed for 10 times and the average of the obtained results is represented in Table 4. The obtained results indicate that although increasing both the number of iterations and population size increase the chance of finding shortest collision-free path, it also increases the algorithm computation time. Therefore, there should be a trade-off between these

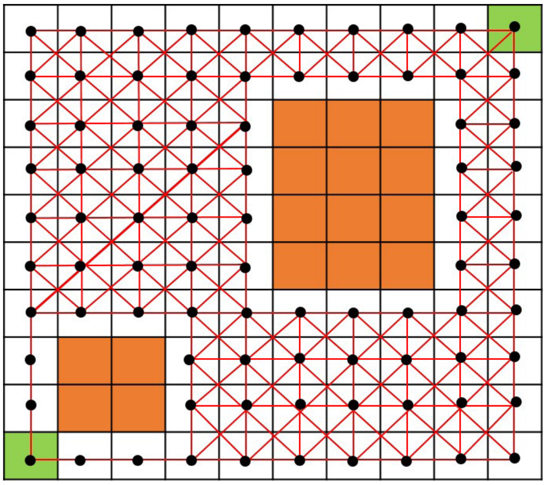


FIGURE 7. Configuration Space to verify effectiveness of G-Ant.

TABLE 4. The simulation results for the configuration space using G-Ant.

No of Iteration	Population Size	Length	Computation Time (Sec)
10	10	21.1923	0.8067
10	40	19.2409	1.0241
10	70	16.9781	1.2141
10	100	16.8610	1.4615
40	10	17.2853	1.6969
40	40	16.6610	2.5589
40	70	16.3782	3.0248
40	100	19.2367	4.3270
70	10	15.9882	2.7683
70	40	16.2568	4.5592
70	70	16.0468	5.6998
70	100	15.7882	7.2338
100	10	16.1882	3.5843
100	40	16.0366	5.4011
100	70	15.7397	7.6876
100	100	15.4225	9.3796

values and computation time of the algorithm. It is worth noting that the number of iterations has more impact than population size on finding shortest collision-free path. For instance, the found path by considering 40 for both number of iterations and population size variables (i.e. length = 16.661) is better than considering 10 and 40 as the values for number of iterations and population size variables, respectively, (i.e. length = 19.2409).

In order to compare the performance of G-Ant with PSO and ACO algorithms a configuration space, which is illustrated in Figures 8 to 13, is utilized. The orange cells represent obstacles. The coordination of start and end points are (0.5, 0.5) and (8.5, 8.5) In this evaluation, the population size is considered constant and equal to 200, while, the number of iterations is assigned to various values includes 200, 400, 600, 800, 1000, and 1200. The obtained results are illustrated through Figures 8 to 13. With the increase of the number of iterations, all approaches find better path, while, when the value of this variable exceed 800, G-Ant, ACO and PSO trapped in local optimum. Although the length of the found path by ACO and PSO is increased after falling in local optimum, this value remains unchanged

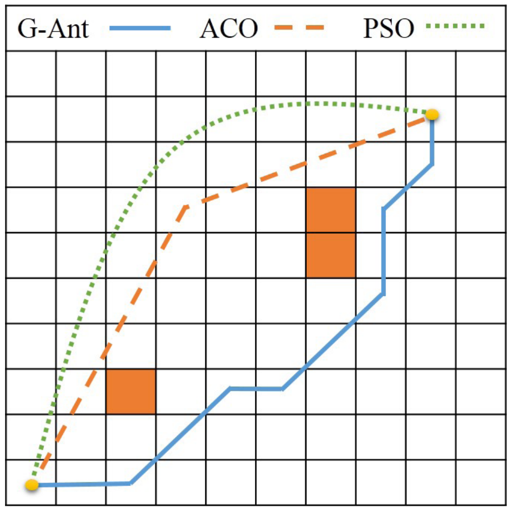


FIGURE 8. Simulation results under 200 iterations.

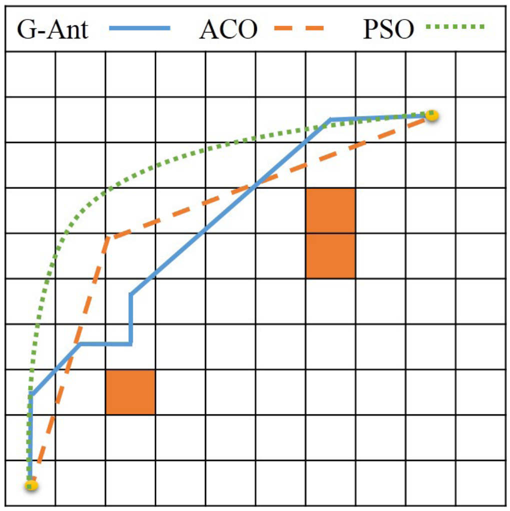


FIGURE 9. Simulation results under 400 iterations.

for G-Ant approach which indicates its robustness against this phenomenon. Moreover, G-Ant outperforms the other approaches considering various number of iterations in terms of travel distance between origin and end points. It is worth noting that PSO provide the smoothest path among the others by eliminating the turns in the found path.

In order to compare the performance of G-Ant with genetic algorithm, three different configuration spaces with different complexities, namely simple, moderate and complex, are designed and illustrated in Figures 14 to 16, respectively. The size of these three configuration spaces are identical and equal to 500×500. The orange cells represent the obstacles.

In all above designed configuration spaces, three different origin and destination points are considered for UGV path planning via genetic algorithm, ACO and G-Ant, and the length of the obtained collision-free path by each approach for three designed configuration spaces is summarized

TABLE 5. Performance evaluation of GA, ACO and G-Ant considering path length.

	Coordination	Start=(5,50), End=(400,450)			Start=(20,55), End=(460,455)			Start=(30,65), End=(465,460)		
Configuration Space Type	**Approach**	GA	ACO	G-Ant	GA	ACO	G-Ant	GA	ACO	G-Ant
Simple	**Length**	618.05	610.34	562.58	659.69	655.22	626.67	652.08	648.62	619.55
Moderate	**Length**	923.04	919.45	910.4	719.48	711.28	700.44	707.09	697.8	667.72
Complex	**Length**	921.03	913.82	913.82	722.28	718.59	710.06	703.79	698.51	674.95

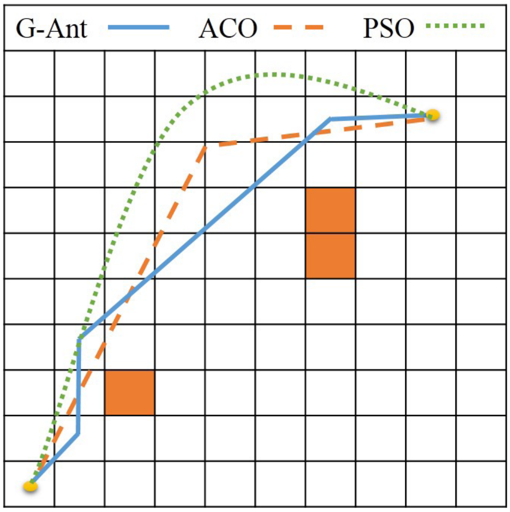


FIGURE 10. Simulation results under 600 iterations.

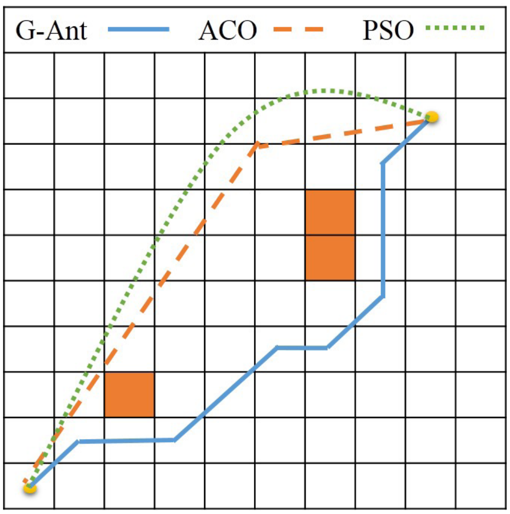


FIGURE 11. Simulation results under 800 iterations.

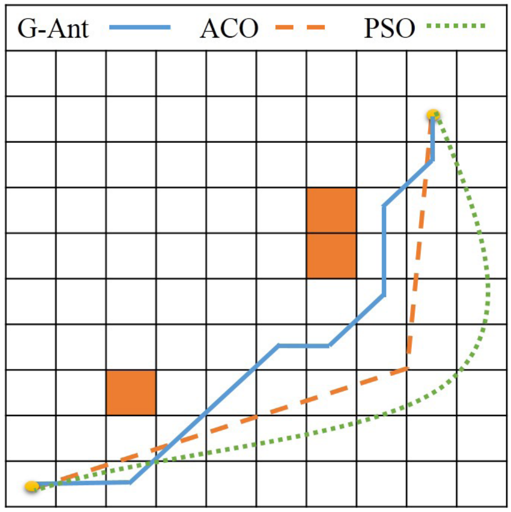


FIGURE 12. Simulation results under 1000 iterations.

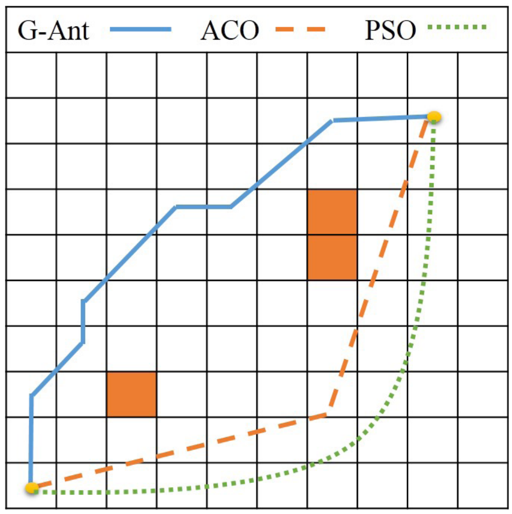


FIGURE 13. Simulation results under 1200 iterations.

in Table 5. Based on the obtained results G-Ant outperforms the other approaches due to considering travel time, travel distance, travel speed and energy consumption, simultaneously, in its path planning procedure. These results prove that ant-based approaches (i.e. ACO and G-Ant) are more robust and suitable for UGV path planning.

These approaches are also evaluated and compared in terms of computation time and number of iterations. The obtained results are summarized in Table 6. Ant-based approaches utilize the state transition rules (i.e. probability function and pheromone update rule) that make these approaches perform more intelligence than genetic-based approaches. In other words, ant-based approaches select the next point via probability function which help the ant agents to select the point near the optimal points and decline unfeasible points. While, genetic-based approaches choose the next

TABLE 6. Performance evaluation of GA, ACO and G-Ant considering computation time, number of iterations and population size.

Configuration Space Type	Simple			Moderate)			Complex		
Approach	GA	ACO	G-Ant	GA	ACO	G-Ant	GA	ACO	G-Ant
Computation Time	1522.03	477.21	671.5	1865.28	648.69	1078	2560.42	972.65	1368.64
No of Iteration	50	30	30	80	45	45	100	50	50
Population Size	20	20	20	35	25	30	50	30	40

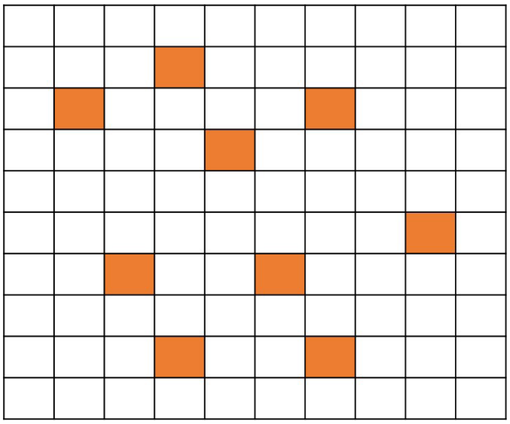


FIGURE 14. Simple configuration space.

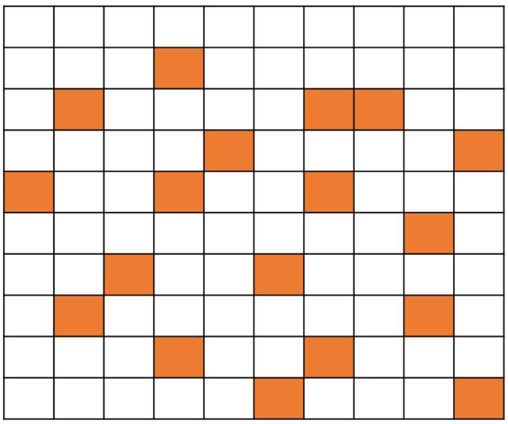


FIGURE 15. Moderate configuration space.

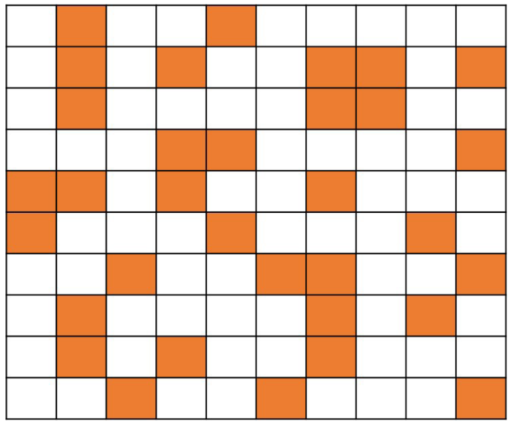


FIGURE 16. Complex configuration space.

point via random approaches which lead genetic algorithm to perform the selection and remove process repeatedly through path planning procedure. Thus as shown in Table 6, ant-based approaches are able to compute the collision-free path quicker with small number of iterations compare to genetic algorithm. As it can be seen, G-Ant has higher computation time compared to ACO. The reason for this is that G-Ant utilizes various factors including travel time, travel distance, travel speed and energy consumption, simultaneously, in its path planning procedure which lead to higher computation time compared to ACO which uses only one route cost (e.g. travel distance, travel time) in its path planning procedure. It is worth noting that the computation time and number of iterations for all approaches increase as the complexity of the configuration space increases.

VI. CONCLUSION

In this paper, a new path planning approach, G-Ant, that combines ant-based algorithm with energy consumption prediction model is proposed and discussed. G-Ant considers the consumed energy in its path planning for UGVs which is one of the challenging issues in UGVs. G-Ant is implemented in MATLAB. The performance of G-Ant was evaluated by comparing it with other existing approaches (i.e. ACO, GA and PSO) under different conditions including different number of iterations, different number of obstacle, and different population size. Travel length, computation time are considered as evaluation metrics. Moreover, three different configuration spaces with different complexities, namely simple, moderate and complex, are designed and utilized in order to compare the performance of G-Ant with genetic algorithm. Based on the obtained results, ant-based approaches (i.e. ACO and G-Ant) find the optimal path much faster than GA in various configuration spaces due to lower computation time as well as number of iterations. Further studies might focus on comparing the consumed energy via various approaches for UGV path planning. In addition, G-Ant should be evaluated in dynamic environment and a method for avoiding collision between UGV and dynamic obstacles should be proposed.

REFERENCES

[1] L. D. Burns, "Sustainable mobility: A vision of our transport future," *Nature*, vol. 497, no. 7448, pp. 181–182, 2013.

[2] S. Le Vine, A. Zolfaghari, and J. Polak, "Autonomous cars: The tension between occupant experience and intersection capacity," *Transp. Res. C, Emerg. Technol.*, vol. 52, pp. 1–14, Mar. 2015.

[3] V. E. Balas and M. M. Balas, "Driver assisting by inverse time to collision," in *Proc. World Autom. Congr.*, 2006, pp. 1–6.

[4] R. W. Button, J. Kamp, T. B. Curtin, and J. Dryden, "A survey of missions for unmanned undersea vehicles," DTIC Document, Tech. Rep., National Defense Res. Inst., 2009. [Online]. Available: http://www.rand.org/content/dam/rand/pubs/monographs/2009/RAND_MG808.pdf

[5] D. M. Tilbury and A. G. Ulsoy, "A new breed of robots that drive themselves," *Mech. Eng.*, vol. 133, no. 2, p. 28, 2011.

[6] A. J. Carfang, E. W. Frew, and D. B. Kingston, "Cascaded optimization of aircraft trajectories for persistent data ferrying," *J. Aerosp. Inf. Syst.*, vol. 11, no. 12, pp. 807–820, 2014.

[7] D. H. Shin and S. Singh, "Path generation for robot vehicles using composite clothoid segments," Robotics Inst., Carnegie Mellon Univ., Tech. Rep. CMU-RI-TR-90-31, Dec. 1990.

[8] P. G. Tzionas, A. Thanailakis, and P. G. Tsalides, "Collision-free path planning for a diamond-shaped robot using two-dimensional cellular automata," *IEEE Trans. Robot. Autom.*, vol. 13, no. 2, pp. 237–250, Apr. 1997.

[9] K. Ioannidis, G. C. Sirakoulis, and I. Andreadis, "Cellular ants: A method to create collision free trajectories for a cooperative robot team," *Robot. Auto. Syst.*, vol. 59, no. 2, pp. 113–127, 2011.

[10] J.-C. Latombe, *Robot Motion Planning*, vol. 124. Berlin, Germany: Springer, 2012.

[11] L. Barnes, M. Fields, and K. Valavanis, "Unmanned ground vehicle swarm formation control using potential fields," in *Proc. Medit. Conf. Control Autom. (MED)*, 2007, pp. 1–8.

[12] W. H. Huang, B. R. Fajen, J. R. Fink, and W. H. Warren, "Visual navigation and obstacle avoidance using a steering potential function," *Robot. Auto. Syst.*, vol. 54, no. 4, pp. 288–299, 2006.

[13] R. Dechter and J. Pearl, "Generalized best-first search strategies and the optimality of A*," *J. ACM*, vol. 32, no. 3, pp. 505–536, 1985.

[14] A. Stentz, "The focussed D* algorithm for real-time replanning," in *Proc. IJCAI*, 1995, pp. 1652–1659.

[15] D. Ferguson and A. Stentz, "Using interpolation to improve path planning: The field D* algorithm," *J. Field Robot.*, vol. 23, no. 2, pp. 79–101, 2006.

[16] J. Von Neumann *et al.*, "Theory of self-reproducing automata," *IEEE Trans. Neural Netw.*, vol. 5, no. 1, pp. 3–14, Jan. 1966.

[17] T.-L. Lee and C.-J. Wu, "Fuzzy motion planning of mobile robots in unknown environments," *J. Intell. Robot. Syst.*, vol. 37, no. 2, pp. 177–191, 2003.

[18] M. F. Selekwa, D. D. Dunlap, D. Shi, and E. G. Collins, Jr., "Robot navigation in very cluttered environments by preference-based fuzzy behaviors," *Robot. Auto. Syst.*, vol. 56, no. 3, pp. 231–246, 2008.

[19] F. Abdessemed, K. Benmahammed, and E. Monacelli, "A fuzzy-based reactive controller for a non-holonomic mobile robot," *Robot. Auto. Syst.*, vol. 47, no. 1, pp. 31–46, 2004.

[20] K. Samsudin, F. A. Ahmad, and S. Mashohor, "A highly interpretable fuzzy rule base using ordinal structure for obstacle avoidance of mobile robot," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 1631–1637, 2011.

[21] S. X. Yang and C. Luo, "A neural network approach to complete coverage path planning," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 718–724, Feb. 2004.

[22] J. Velagic, N. Osmic, and B. Lacevic, "Neural network controller for mobile robot motion control," *Int. J. Intell. Syst. Technol.*, vol. 3, no. 2, pp. 127–133, 2008.

[23] M. K. Singh and D. R. Parhi, "Intelligent neuro-controller for navigation of mobile robot," in *Proc. Int. Conf. Adv. Comput., Commun. Control*, 2009, pp. 123–128.

[24] V. Castro, J. Neira, C. Rueda, J. Villamizar, and L. Angel, "Autonomous navigation strategies for mobile robots using a probabilistic neural network (PNN)," in *Proc. 33rd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Nov. 2007, pp. 2795–2800.

[25] F. M. Marchese and M. Dal Negro, "Path-planning for multiple generic-shaped mobile robots with MCA," in *Proc. Int. Conf. Comput. Sci.*, 2006, pp. 264–271.

[26] G. Liu, T. Li, Y. Peng, and X. Hou, "The ant algorithm for solving robot path planning problem," in *Proc. 3rd Int. Conf. Inf. Technol. Appl. (ICITA)*, 2005, pp. 25–27.

[27] Y. Cen, C. Song, N. Xie, and L. Wang, "Path planning method for mobile robot based on ant colony optimization algorithm," in *Proc. 3rd IEEE Conf. Ind. Electron. Appl.*, Jun. 2008, pp. 298–301.

[28] K. Gopalakrishnan, S. Ramakrishnan, and C. Dagli, "Optimal path planning of mobile robot with multiple targets using ant colony optimization," in *Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 16. New York, NY, USA: ASME, 2006.

[29] M. A. P. Garcia, O. Montiel, O. Castillo, R. Sepúlveda, and P. Melin, "Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation," *Appl. Soft Comput.*, vol. 9, no. 3, pp. 1102–1110, 2009.

[30] H. Mei, Y. Tian, and L. Zu, "A hybrid ant colony optimization algorithm for path planning of robot in dynamic environment," *Int. J. Inf. Technol.*, vol. 12, no. 3, pp. 78–88, 2006.

[31] N. B. Sariff and N. Buniyamin, "Ant colony system for robot path planning in global static environment," in *Proc. 9th WSEAS Int. Conf. Syst. Sci. Simulation Eng. (ICOSSSE)*, 2010, pp. 192–197.

[32] A. C. Nearchou, "Path planning of a mobile robot using genetic heuristics," *Robotica*, vol. 16, no. 5, pp. 575–588, 1998.

[33] C. Liu, H. Liu, and J. Yang, "A path planning method based on adaptive genetic algorithm for mobile robot," *J. Inf. Comput. Sci.*, vol. 8, no. 5, pp. 808–814, 2011.

[34] B. B. V. L. Deepak and D. Parhi, "Intelligent adaptive immune-based motion planner of a mobile robot in cluttered environment," *Intell. Service Robot.*, vol. 6, no. 3, pp. 155–162, 2013.

[35] M. Yuan, S.-A. Wang, C. Wu, and N. Chen, "A novel immune network strategy for robot path planning in complicated environments," *J. Intell. Robot. Syst.*, vol. 60, no. 1, pp. 111–131, 2010.

[36] P. K. Mohanty and D. R. Parhi, "Cuckoo search algorithm for the mobile robot navigation," in *Proc. Int. Conf. Swarm, Evol., Memetic Comput.*, 2013, pp. 527–536.

[37] X.-D. Liang, L.-Y. Li, J.-G. Wu, and H.-N. Chen, "Mobile robot path planning based on adaptive bacterial foraging algorithm," *J. Central South Univ.*, vol. 20, no. 12, pp. 3391–3400, 2013.

[38] L. Lu and D. Gong, "Robot path planning in unknown environments using particle swarm optimization," in *Proc. 4th Int. Conf. Natural Comput.*, vol. 4. 2008, pp. 422–426.

[39] C.-F. Juang and Y.-C. Chang, "Evolutionary-group-based particle-swarm-optimized fuzzy controller with application to mobile-robot navigation in unknown environments," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 2, pp. 379–392, Apr. 2011.

[40] S. H. Tang, F. Kamil, W. Khaksar, N. Zulkifli, and S. A. Ahmad, "Robotic motion planning in unknown dynamic environments: Existing approaches and challenges," in *Proc. IEEE Int. Symp. Robot. Intell. Sensors (IRIS)*, Oct. 2015, pp. 288–294.

[41] Q. Zhu and L. Wang, "A new algorithm for robot path planning based on scout ant cooperation," in *Proc. 4th Int. Conf. Natural Comput.*, vol. 7. 2008, pp. 444–449.

[42] Jayadeva, S. Shah, A. Bhaya, R. Kothari, and S. Chandra, "Ants find the shortest path: A mathematical proof," *Swarm Intell.*, vol. 7, no. 1, pp. 43–62, 2013.

[43] M. Dorigo and T. Stützle, "Ant colony optimization: Overview and recent advances," in *Handbook of Metaheuristics*. Berlin, Germany: Springer, 2010, pp. 227–263.

[44] M. Dorigo, "Optimization, learning and natural algorithms," Ph.D. dissertation, Politecnico di Milano, Milan, Italy, 1992.

[45] M. Dorigo and T. Stutzle, "Ant colony optimization: Overview and recent advances," IRIDIA, Univ. Libre de Bruxelles, Bruxelles, Belgium, Tech Rep. 2009-013, 2009.

[46] A. Sadrpour, J. Jin, and A. G. Ulsoy, "Mission energy prediction for unmanned ground vehicles," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2012, pp. 2229–2234.

[47] A. Sadrpour, J. Jin, and A. G. Ulsoy, "Mission energy prediction for unmanned ground vehicles using real-time measurements and prior knowledge," *J. Field Robot.*, vol. 30, no. 3, pp. 399–414, 2013.

[48] M. Vojtíšek, M. Fenkl, and M. Dufek, "Effect of high-speed, performance driving on exhaust emissions of modern light-duty vehicles," in *Advanced Automation Engineering*, P. Novotnỳ, Ed. Brno, Czech Republic: Tribun EU, 2008.

[49] I. M. Berry, "The effects of driving style and vehicle performance on the real-world fuel consumption of US light-duty vehicles," Ph.D. dissertation, Dept. Mech. Eng., Massachusetts Inst. Technol., Cambridge, MA, USA, 2010.

[50] J. Van Mierlo, G. Maggetto, E. Van de Burgwal, and R. Gense, "Driving style and traffic measures-influence on vehicle emissions and fuel consumption," *Proc. Inst. Mech. Eng. D, J. Automobile Eng.*, vol. 218, no. 1, pp. 43–50, 2004.

[51] I. De Vlieger, D. De Keukeleere, and J. Kretzschmar, "Environmental effects of driving behaviour and congestion related to passenger cars," *Atmos. Environ.*, vol. 34, no. 27, pp. 4649–4655, 2000.

[52] J. N. Barkenbus, "Eco-driving: An overlooked climate change initiative," *Energy Policy*, vol. 38, no. 2, pp. 762–769, 2010.

[53] A. A. Sadrpour, "Acceptance testing and energy-based mission reliability in unmanned ground vehicles," Ph.D. dissertation, Dept. Ind. Oper. Eng., Univ. Michigan, Ann Arbor, MI, USA, 2014.

[54] L. Ljung, "System identification," in *Signal Analysis and Prediction*. Berlin, Germany: Springer, 1998, pp. 163–173.

[55] J. M. Lucas and M. S. Saccucci, "Exponentially weighted moving average control schemes: Properties and enhancements," *Technometrics*, vol. 32, no. 1, pp. 1–12, 1990.

[56] T. M. Howard, *Adaptive Model-Predictive Motion Planning for Navigation in Complex Environments*. Pittsburgh, PA, USA: Carnegie Mellon Univ., 2009.

[57] J. Hardy and M. Campbell, "Contingency planning over probabilistic obstacle predictions for autonomous road vehicles," *IEEE Trans. Robot.*, vol. 29, no. 4, pp. 913–929, Aug. 2013.

[58] U. Lee, S. Yoon, H. Shim, P. Vasseur, and C. Demonceaux, "Local path planning in a complex environment for self-driving car," in *Proc. IEEE 4th Annu. Int. Conf. Cyber Technol. Autom., Control, Intell. Syst. (CYBER)*, Jun. 2014, pp. 445–450.

[59] P. Zhao, J. Chen, T. Mei, and H. Liang, "Dynamic motion planning for autonomous vehicle in unknown environments," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 284–289.

[60] A. Broggi, P. Medici, P. Zani, A. Coati, and M. Panciroli, "Autonomous vehicles control in the VisLab intercontinental autonomous challenge," *Annu. Rev. Control*, vol. 36, no. 1, pp. 161–171, 2012.

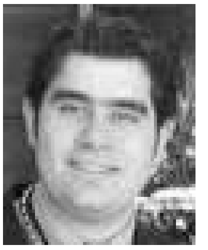
[61] M. McNaughton, C. Urmson, J. M. Dolan, and J.-W. Lee, "Motion planning for autonomous driving with a conformal spatiotemporal lattice," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2011, pp. 4889–4895.

[62] M. Pivtoraiko, R. A. Knepper, and A. Kelly, "Differentially constrained mobile robot motion planning in state lattices," *J. Field Robot.*, vol. 26, no. 3, pp. 308–333, 2009.

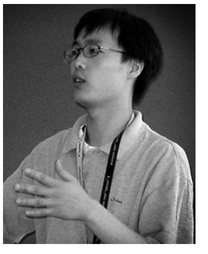
[63] D. C. Montgomery, *Introduction to Statistical Quality Control*. Hoboken, NJ, USA: Wiley, 2007.

[64] M. R. Jabbarpour *et al.*, "Ant-based vehicle congestion avoidance system using vehicular networks," *Eng. Appl. Artif. Intell.*, vol. 36, pp. 303–319, Nov. 2014.

MOHAMMAD RZEA JABBARPOUR received the B.Sc. degree in information technology engineering from Tabriz, Tabriz University, Iran, and the M.Sc. and Ph.D. degrees in computer science with data communication and computer network specialty from the University of Malaya, Malaysia, in 2012 and 2015, respectively. His Ph.D. was sponsored by High Impact Research, the most prestigious scholarship in Malaysia, funded by the Ministry of Education. He is currently serves as a Post-Doctoral Fellowship with the Iran Telecommunication Research Center, Iran. His current research interests include vehicular and mobile ad hoc network (Architectures, Protocols, Security, and Algorithms), vehicular cloud computing, vehicle traffic routing systems, artificial intelligence, and swarm intelligence. He received various awards, including the WiC 2015 Asia Invention Association grand award, Gold medal, the PECIPTA 2015 Bronze Medal, and the Singapore Challenge Merit award for his patented idea. While he has published various journal and conference papers, he is serving as a member of the Editorial Board and a Reviewer of international journals and conferences.



HOUMAN ZARRABI received the Bachelor of Computer Engineering degree from the National University of Iran (also known as Shahid Beheshti University) in 2003, and the master's and Ph.D. degrees from the Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada, in 2006 and 2011, respectively. He has been affiliated with the Department of Electrical Engineering, Ecole Polytechnique de Montreal. His research interests include energy management, VLSI, green computing, embedded systems, wireless sensor networks, and IoT.



JASON J. JUNG received the B.Eng. degree in computer science and mechanical engineering and the M.S. and Ph.D. degrees in computer and information engineering from Inha University in 1999, 2002, and 2005, respectively. He was a Visiting Scientist with the Fraunhofer Institute, Berlin, Germany, in 2004, and a Post-Doctoral Researcher with INRIA Rhone-Alpes, France, in 2006. He has been an Assistant Professor with Yeungnam University, South Korea, since 2007. He has been an Associate Professor with Chung-Ang University, South Korea, since 2014. His research topics are knowledge engineering on social networks by using many types of AI methodologies, such as, data mining, machine learning, and logical reasoning. Recently, he have been working on intelligent schemes to understand various social dynamics in large-scale social media, such as Twitter and Flickr.

PANKOO KIM received the B.E. degree from Chosun University, Gwangju, South Korea, in 1988 and the M.S. and Ph.D. degrees in computer engineering from Seoul National University, Seoul, South Korea, in 1990 and 1994, respectively. He is currently a Full Professor with Chosun University. His specific interests include semantic web techniques, semantic information processing and retrieval, multimedia processing, semantic web, and system security. He is the Editor-in-Chief of the *IT CoNvergence PRActice Journal*.