# Collaborative Evolutionary Planning Framework (EPF) for Route Planning

Kashif Zafar National University of Computer & Emerging Sciences A. K. Brohi Road, H-11/4, Islamabad Abdul Rauf Baig National University of Computer & Emerging Sciences A. K. Brohi Road H-11/4, Islamabad Ayesha Khan National University of Computer & Emerging Sciences A. K. Brohi Road H-11/4, Islamabad

# ABSTRACT

This research presents a collaborative evolutionary planning framework for large scale grid exploration and planning problems. It caters for both dynamic and unknown environments using evolutionary techniques. In addition, we integrate the exploration and planning process in a unified framework using multi agent system. As a proof of success, we have developed extensive simulation with realistic obstacles and target. Our algorithm addresses the issues involved during such exploration and post exploration route planning. It acts as a controller and navigator for multiple agents and demonstrates the applicability for two different domains, Field Exploration and Route Planning. The EPF uses an optimized search algorithm for exploration phase and genetic algorithm for optimization of route in dynamic environments. The EPF can be used in different exploration and route planning problems but this paper focuses on obstacle detection and avoidance for its implementation.

# **General Terms**

Evolutionary Algorithm, Optimization

# **Keywords**

Genetic Algorithm, Agents, Planning, Autonomous agent, Route Planning

# **1. INTRODUCTION**

The route planning algorithms [2] are considered the startgoal problem where the objective is to find a route between two points. The conventional route planning algorithms are unable to address specific applications like floor cleaning [3], mine detection [4], lawn mowing [5], etc. These applications require a robust field coverage route planning algorithm that uses sensor-based information for planning, where sensors are the electronic devices used to sense and build the perception of the environment. The environment can be known, partially known or unknown. It can also be static or dynamic depending upon the application area. The static environment is the environment in which goal remains stationary while in dynamic environment, goal can change during route planning.

Most of the planning algorithms deal with grids of data, called maps, which store the cost of each node, or point on the grid. These algorithms try to find a path along the grid (moving in only the four cardinal directions, or the four cardinal directions as well as the four diagonals) with the lowest total cost (e.g. distance). This research presents a Collaborative EPF for route planning and has been tested for mine detection problem and unifies mine exploration and post exploration route planning. The offline planners are inflexible and lack scalability but when they are combined with online planning, they are flexible, scalable and adaptive and can be used for dynamic and unknown environments. The online planning is the real time planning process that can cater uncertainty and the issue of adaptability. The major contribution of this paper is a unified framework for route planning and navigation along with obstacle avoidance capabilities.

The Section 2 discusses the work related with exploration and coordination. Section 3 will discuss the system architecture of a collaborative evolutionary planning framework and Section 4 will discuss implementation details and experiments performed. The Section 5 presents the results and comparisons and Section 6 gives conclusions.

# 2. LITERATURE REVIEW

The basic styles of exploration methods have emerged which rank unexplored areas and make rational choices are: frontier based [6], A\* algorithm [7], D\* algorithm [8], LRTA\* [9], PBLRTA\* [10] and generalized Voroni graph method [11]. All of them use behaviors for navigation, but are different in how they set the navigational goals. Following is provided a simplified overview of each method that can be used for the exploration of an unknown region. Frontier-based exploration was pioneered by Brian Yamauchi [2]. The robots can navigate using maps, some robots are equipped to update their own maps and some are equipped to build their own maps. A given territory is required to be mapped in advance undergoing a mapping process, providing either the exact locations of obstacles as in the case of metric maps or a graph representing the connectivity between open regions as in the case of topological maps. As a result, most mobile robots become unable to navigate efficiently when placed in unknown environments.

Exploration has the potential to enable robots to rule out this limitation. *Exploration* is an act of navigating through an unknown environment while constructing a map that can be used for subsequent navigation. A plausible exploration is one that results in generating a complete or nearly complete map in reasonable amount of time.

collaborative behavior. The simulation used has been implemented using C# language and in Microsoft.Net



Figure 1. Collaborative Evolutionary Planning Framework

The Frontier Based Exploration uses hops to select where to move the robot in its next step, so that an unexplored area of the map may be explored, and it tries to do it in an optimum fashion. If a robot with a perfect map could navigate to a particular point in space, that point is considered accessible.

The Hybrid Frontier Based Architecture [10] has been used previously to address the issue of field exploration and mine detection. It demonstrates the exploration of an unknown environment by a group of autonomous agents using Frontier-based exploration technique.

# 3. COLLABORATIVE EVOLUTIONARY PLANNING FRAMEWORK

The Collaborative EPF presents a unified framework for exploration and post exploration route planning in dynamic and unknown environments. This research combines exploration and planning for the first time and compares the results with traditional planners for performance and efficiency.

The Collaborative EPF consists of two phases i.e. exploration phase and planning phase as shown in Figure 1. The exploration phase is responsible for obstacle detection along with building perception map. Planning phase consists of two sub modules i.e. evolutionary planner and cooperative planner. This phase caters the requirements of dynamic environments by using evolutionary algorithm along with implementing cooperative planning for framework. Below is the explanation of the two phases of Collaborative EPF.

#### 3.1 Exploration Phase

After getting the sensory information, the exploration phase starts exploration of the environment along with implementing obstacle avoidance. In order to enhance the efficiency of the exploration phase, multiple agents can be used with coordination capabilities. They share and exchange sub maps for perception building. This phase requires robust coordination module for information exchange and communication as shown in Figure 2. The coordinating module of an agent is responsible for this activity.

The anatomy of an agent is composed of its belief and its capabilities. The belief of an agent is represented as a two dimensional occupancy grid. Each cell of the occupancy grid symbolizes a concept of environment i.e. free cell, obstacle cell, and unknown cell. The agent applies its deliberation capability on this belief in order to infer a goal, which it will later pursue.



Figure 2. Basic coordination architecture

The agent has the following set of capabilities, which enable it to explore a given unknown environment:

- 1. Sense
- 2. Mapping
- 3. Localization
- 4. Coordinate

The agent's sensory mechanism is represented as footprint of eight adjacent cells to its current position. Whenever an agent performs a sensory sweep it gains new information about its environment. The agent has mapping capability implying the agent updates its current belief i.e. the occupancy grid by fusing information gained about new cells as a resultant of the sensory sweep and the agent's belief gets subsequently updated.

The agent is capable of localizing itself with respect to its current belief i.e. current mapped territory. Localization implies the re-location of the agent to a new location in order to explore unknown environment. The Localization function is an implementation of a frontier based coverage technique [1]. The Localization process involves the following sub steps to deduce a goal location:

- 1. Retrieval of frontier cells
- 2. Scan the traversed area for any unexplored cells
- 3. Calculation of cost to each frontier cell from current location
- 4. Declaration of least cost cell or frontier cell as goal cell

Frontier cells are identified by scanning all visited free cells, a visited free cell is declared a frontier cell if it is adjacent to at least one unknown cell in the horizontal or vertical direction. The cost of each frontier cell is calculated using a path navigator instead of blind search method. The path navigator is configured with Manhattan distance as the heuristic function.

The Manhattan distance [12] is total number of cells moved horizontally and vertically to reach the target cell from the current cell, ignoring diagonal movement, and ignoring any obstacles as shown in equation 1.

$$h(n) = D * (abs(n.x-goal.x) + abs(n.y-goal.y)) - (1)$$

Where h(n) is the heuristic estimate function used and h value guides search toward heuristically promising states. The Frontier cell with the least cost is declared as the next goal state for the agent.

#### **3.2. Planning Phase**

Once the exploration phase is completed, the role of planning phase comes in. The planning phase is responsible for planning the task execution. As Collaborative EPF addresses the issues involved in dynamic environment, so a robust and scalable planner is required.

Primarily, the agents search the environment and build initial perception by using their sensory sweep and activities of the exploration phase. After the execution of exploration phase, the agents have the complete map of the environment with information regarding obstacles and other such hazardous points. This initial map can be considered as static and it becomes an input for the next phase. In planning phase, there are two main modules, i.e. Evolutionary Planner and Cooperative Planner. These two modules work together in close contact for the planning process.

#### 4. EXPERIMENTATION

The experimental setup consists of 5 different size grids for performance evaluation. The grids range in size from 20x20, 40x40, 60x60, 80x80, and 100x100. The option of using agents ranges from 1 to 6 agents. The maps are also varied by number of obstacles and number of mines. The experiments used 5 different maps generated randomly for each grid size and performed the experiments by taking time, distance, and number of cells traversed, number of mines detected and number of obstacles identified as the parameters.

The experiments were performed by taking the average time for 10 runs of each algorithm. Primarily, two different algorithms were tested and compared in the experiments and graphs were drawn for average time for each grid size and cells traversed by each algorithm.

It has three major components i.e. load environment, exploration phase and planning phase. The main task of the load environment phase is to load the maps according to specific grid size selected by the user. The maps generated randomly for each grid. After the selection of the grid size and appropriate map, the map appears in the global map window. The local map window shows the perception of the agents view. After the loading phase, agent's view appears to be black as agent has not stared exploration. The agent starts exploration from a random starting state to test its performance and provision for the selection of the starting point is also provided. The server starts automatically and configures the log file accordingly. The option of selecting number of agents is provided and we have to select the number of agents before loading the environment. The Figure 3 shows the loaded map in the global window.



Figure 3. Global window with loaded map

The exploration phase gives two options for the exploration algorithm. The exploration can be performed by Frontier based algorithm as well as the learning real time  $A^*$  algorithm for comparison. The exploration phase starts according to the number agents selected.



Figure 4. Two agents exploring the map

When multiple agents are used, the simulation runs by using the coordination module. The server manages the explored and unexplored region along with broadcasting sub maps to each peer agent. The communication between each agent is controlled by the coordination module embedded in each exploring agent. Each agent is highlighted by using a different color. A function is incorporated to trace the step by step execution of the simulation as shown in Figure 4 & 5.



Figure 5. Exploration with Coordinating Multi Agents

After the successful operation of the exploration phase, the next objective is to device an optimal route. The provision for the start point and the destination point is also given in the simulation. The resultant path will be an optimal route after avoiding obstacles and mines as shown in Figure 6.



Figure 6. Post Exploration Route Planning

#### 5. RESULTS

The two phases of the implementation was evaluated separately, i.e., the field exploration and the route planning.

The exploration phase uses multi agent system with agents for searching. The simulation was tested by two separate algorithms i.e. Frontier based exploration and Learning real time A\* algorithm. The performance of the two algorithm has been tested with different number of agents and LRTA\* showed much better results as compared to Frontier based algorithm. Five different maps with different number of mines and obstacles were used for the experiment and they were generated randomly for each run. The Figure 7 shows the distance traversed by agents i.e. cells traversed during exploration. Each data value has been taken by running each experiment 10 times and average was taken for each map.





Figure 7. Distance traversed by frontier based exploration

The increase in number of agents reduces the distance and improves the performance of the EPF framework. A slight increase has been noticed in the distance traversed for 6 agents due to communication and processing over heads. While in all the graphs the results are consistent with the objectives of the EPF framework and proved to be a successful approach for multi agent exploration problems. The time for exploration was also checked and it was consistent with the graphs for distance.

The average time for exploration reduces exponentially by increasing the number of agents. The time reduces drastically for the exploration and it shows the performance of the Collaborative EPF for large scale grid exploration problems. The average time has been calculated by taking the average of 10 runs for each algorithm on a particular grid and for a particular map. Another parameter is the number of cells traversed during exploration. Again average of 10 runs was taken for the measure and results were promising and showed the performance of the framework. The x-axis for each graph shows the number of agents and y-axis shows the distance traversed. The EPF framework can be used for time critical multi agent applications for performance. The reduction in time with multiple agents clearly indicates the possibility of using multiple agents for large exploration problems. We have used 1, 2, 4, & 6 agents for each map and results obtained were compared by using two different algorithms for accuracy.

#### 6. CONCLUSIONS

This paper presents a Collaborative Evolutionary Planning Framework for integrating exploration and planning modules in a unified framework. It provides a robust, scalable and reliable framework for multi agent tasks. This paper uses the mine detection and route planning for its testing and results showed that this framework performs well with multiple agents. The reduction in time and distance traversed by using multiple agents clearly indicates the avenue of integration of searching and planning modules. The exploration module implements and compares frontier based exploration and LRTA\* algorithms. The performance of LRTA\* has been better as compare to Frontier based exploration. This research concludes by providing a statistical comparison of results i.e. distance traversed and exploration time for different environment configurations i.e. varying maps and number of agents. The evolutionary programming techniques have shown better results for dynamic environments and can be further tested for different exploration and planning problems.

# 7. REFERENCES

[1] B. Yamauchi "A Frontier Based Approach for Autonomous Exploration" *IEEE International Symposium* on Computational Intelligence on Robotics and Automation, 1997

[2] B. Yamauchi "Frontier Based Exploration using Multi Robots" International Conference on Autonomous Agents, 1998

[3] D. Latimer, et al, "Towards sensor based coverage with robot teams" *IEEE International Conference on Robotics* and Automation, 2002, pp. 961-967

[4] A. Howard, M. J. Mataric and G. S. Skhatme, "Mobile sensor network deployment using potential fields: a distributed, scalable solution to the Area coverage

problem," 6th International Symposium on Distributed Autonomous Robotics Systems, 2002, pp. 299-308

[5] A. Solanas and M. Angel "Coordinated Multi-Robot Exploration through Unsupervised Clustering of Unknown Space"

[6] A. Stentz "Optimal and Efficient Path Planning for Partially-Known environments" In Proceedings of the IEEE International Conference on Robotics and Automation, May 1994

[7] Optimal path planning in partially known environments (1992)

[8] S. Kamio and H. Iba "Random Sampling Algorithm for Multi-agent Cooperation Planning"

[9] Xiao, Michalewicz, Zhang, and Trojanowski "Adaptive Evolutionary Planner/Navigator for Mobile Robots" *IEEE Transactions on Evolutionary Computations 1996* 

[10] K Zafar, A. R. Baig, S. Badar "Mine Detection and Route Planning in Military Warfare using Multi Agent System" *IEEE COMPSAC conference ESAS 2006* 

[11] S. Ahn, N. L. Doh, K. Lee, W. K. Chung "Incremental and robust construction of Generalized Voronoi Graph (GVG) for mobile guide robot" *In Proceedings of Intelligent Robots and Systems, IROS 2003*