IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 09, Issue 7 (July. 2019), ||S (II) || PP 43-60

Adaptive Optimal Task Assignment for Cooperative Autonomous Vehicles

Abdel rahman M. Ezz[1,*], AlaaKhamis[2], Yasser G. Hegazy[3]
[1]*Mechatronics department, Faculty of Engineering and Materials Science, German University in Cairo (GUC), Egypt*
[2]*Centre for Pattern Analysis and Machine Intelligence (CPAMI), University of Waterloo, Waterloo, ON, Canada*
[3]*Faculty of Information Engineering and Technology, German University in Cairo (GUC), Egypt*
Corresponding Author: Abdelrahman M. Ezz

Abstract: Natural catastrophes and man-made disasters reported worldwide are steadily increasing each year in frequency and destructive capacity. Timely information, early warning of potential hazards, effective search andrescuetechniquesarerequiredtomitigatetherisksassociatedwiththesedisastersandreducetheirdestructiveconsequ ences.Thus,it was not surprising in the technological era we live in nowadays to include the search and rescue applications in the agenda of robotics research field. Consequently, the aim of this paper is to study the applicability of cooperative autonomous vehicles in search and rescue missions. The main focus of the pa- per is adaptive optimal task allocation of these vehicles as one of the main challenging problems of multi-robot systems. In the context of this paper, the problem of task assignment is handled as a multi-criteria optimization problem. A comparative study between stochastic (Particle Swarm Optimization-PSO)and deterministic greedy (Hungarian and modified Hungarian algorithms) optimization solvers is conducted. An adaptation mechanism is also proposed to handle the environment changes and possible failures of vehicles. State-of-the-art robotics middleware-frame work Robot Operating System(ROS)andGazebo3Dsimulatorareusedtosimulatearealistic search and rescue mission in two scenarios that differ in communication capabilities; global and local scenarios. Access to global information is guaranteed in global information scenario. While in the local scenario, failure in the global communication network occurs, thus vehicles have to rely on themselves to initiate an ad-hoc communication relaying network. Two architectures; centralized and distributed, are used in implementing the localscenario

Keywords: Multi-robot systems, Optimization, Particle swarm, Hungarian algorithm, Autonomous vehicles

Date of Submission: 11-07-2019 Date of acceptance: 26-07-2019

I. INTRODUCTION

The United Nations office for disaster risk reduction (UNSIDR) reported 327 disasters, of which 191 were natural disasters and 136 man-made disasters [1]. Some victims could be trapped inside the disaster area because of route blocking and panic emotion. The only way for these victims to survive may be waiting for help from the rescue team. But due to uncertainty of the accurate positions of the victims and their physical status, the search team may take too long to discover them. Also in disasters like explosions caused in nuclear sites and hazardous environments, it's unsafe for rescue team of humans to enter the disaster's area for a long time. This can cause serious harm to humans. Moreover humans could not be able to cover a large area of a disaster. Consequently humans may be inefficient for search and rescue missions [2].

Therefore, it was not surprising, in the technological era we live in now, to use the robots in the search and res- cue missions. The research field of robotics, in the last few years, was interested in the challenging problems need to be faced in order to use robots in many civilian and military applications like search and rescue missions to reduce the loss of humans. Also the attention of the research field of robotics was directed towards using multi-robot system (MRS) instead of using one single robot, due to its significant advantages over the single robots [3]. Multi-robot systems can do complex tasks with high efficiency, reliability and in short-time [4] [5]. Multi-robot system are resolving task complexity by subdivision of complex tasks into smaller ones. Moreover, MRS is simpler in design compared to a single robot, which need to be well designed with a lot of capabilities in it toper form a mission. In contrast MRS consists of group of robots so each one should be capable of just one or two things. Furthermore, it is more efficient in terms of time where, MRS have a group of robots that works in parallel. In addition to, reliability of the system is increased since in MRS when a robot failed, the remaining robots can continue the mission. In contrast when a single robot faces a failure the whole mission is failed[3].

The objective of this paper is to handle the task assignment problem for cooperative autonomous vehicles in realistic simulation of search and rescue missions and to use autonomous vehicles with heterogeneous capabilities in order to satisfy the needs of trapped victims as much as possible. In this study different types of solvers are applied to solve the task assignment problem as stochastic and deterministic solvers. Furthermore, the objective continues to implement an adaptation entity to adapt the task assignment made for cooperative vehicles according to the environmental perceptions. Finally, the proposed approaches are tested on different scenarios. These scenarios differ in the communication capabilities whether the vehicles have access to global information to communicate with each others and with the base station or not. Moreover, test scenarios differ in coordination architecture which could be centralized or distributed. The simulation environment used in this paper is implemented in a state-of-the-art middleware-framework called robot operating system (ROS) and simulated in Gazebo 3D simulator. The contribution is mainly about implementation a realistic scenarios for search and res- cue missions in disaster areas using cooperative autonomous vehicles in Robot Operating System (ROS). Moreover, these scenarios investigate the problem of optimal task assignment under different communication capabilities and different organizational architectures. Finally, a comparison between stochastic and deterministic optimization techniques was conducted.

The remainder of the paper is organized as follows. Section 2 provides a detailed literature review on the optimal assignment problem from the scope of cooperative autonomous vehicles. Section 3 presents the optimal assignment problem, its modelling and formulation followed by describing of the proposed approaches in Section 4.Experimental results are presented and discussed in Section 5. Finally, conclusion and future work are summarized in Section 6.

II. LITERATUREREVIEW

Task assignment problem is a common challenging problem in many disciplines not only in autonomous vehicles. It can be formulated with many approaches. Some of these approaches are: the multiple traveling salesmen problem (mTSP). It can be considered as a general form of traveling salesman problem (TSP), which is used as a benchmark problem for a wide range of discrete optimization problems. It can be defined as an algorithmic problem tasked with finding the shortest route between a set of points and locations that must be visited. In the problem statement, the points are the cities a salesperson might visit. The salesman‘s goal is to keep both the travel costs and the distance traveled as low as possible. Multiple travelling salesman problem has been used with different variation to formulate task assignment for vehicles systems [6] [7] [8] [9]. Moving on to another approach, discrete fair division which is a mathematical theory aims to fairly divide re- sources or goods among several individuals. The fairness in this theory can have various types. It’s one of the commonly used approaches to formulate task assignment [10] [11] [12] [13]. Furthermore, task assignment problem for autonomous vehicles systems can be modeled as an optimal assignment problem (OAP). OAP is a problem where a number of agents (m) and a number of jobs (n) are given. Each job requires one agent and each agent is searching for a job to do. Many researches used the formulation of task assignment as optimal assignment problem [14] [15] [16] [17] [18]. Particle swarm optimization is used as a stochastic solver to the optimal assignment problem. It is an effective technique to solve the optimal assignment problems in robotics [19] [20] [21] [22]. For example, orthogonal PSO which is implemented for solving intractable large parameter optimization problems such as optimal assignment problems [23] and hybrid form of PSO is used to solve task assignment problem in [24]. The deterministic algorithm chosen to solve the assignment problem is the Hungarian-based optimization algorithm. Hungarian algorithm is a combinatorial optimization algorithm which was developed by Harold W. Kuhn in 1955 to solve assignment problem [25]. Moreover, Hungarian algorithm was based on the work of two Hungarian mathematicians, Dénes Ko˝nig and Jeno˝ Egerváry, so it was named by this name [26]. The Hungarian algorithm has been used to solve task allocation problems for multi-vehicle systems [27][28][29][30]. Task assignment problem was studied in both centralized and decentralized architecture as described in the following subsections.

III. Centralized Architecture

In centralized architecture system, the system includes a central unit or central agent responsible for all decision making.Thiscentralunitshouldbeconnectedtoallthevehiclesincludedinthesystemtobeabletocollectdata aboutthevehiclesateveryinstant.Thisdataisrequiredtotakedecisionsandformonitoringtheperformanceof the system. Also, it gets the data about the environment sensed by the vehicles in the system. All the algorithms that are related to cooperative behavior between the vehicles should be implemented in this central unit. IT collects the data, process, analyze and take decisions then send it to the vehicles through the communication network between them to control their actions [31].

In [32], the authors claimed that the centralized architecture is commonly used in research as well as many industrial applications to that it’s simpler in its implementation than other approaches. Conversely in [33], Elmogy et al stated that the centralized architecture is inefficient in terms of system scalability, where the system performance decreases with increasing number of vehicles due to increasing in computational time. Although centralized architecture is simple in implementation, it has a disadvantage presented in the reliability of the system in situations of system failure or malfunction. In such cases, if the central agent fails, this will lead to complete failure of the whole system until the replacement or fixation of this central agent. [34], Authors in [31] presented the use of centralized architecture in order to assign tasks to a system of mobile robots.

IV. Decentralized Architecture

Decentralized architecture is an architecture between agents or vehicles in the system, where there is no central controller or central agent but its role is divided upon the agents in the system. The division between the vehicles or the agents of the system depends on the communication between them. Moreover, the Communication network between vehicles can be a global network where all the vehicles can communicate with each other. This configuration is found in the centralized architecture. Conversely, the global information may be unavailable for the vehicles. In this case they need to establish a communication network in order to communicate. This configuration is valid only on the neighborhood level where the vehicles can only communicate in their communication range.

One of the advantages ofdecentralizedarchitectureisthehighscalabilityofthesystem.Thesystemcanaddor remove vehicle without leaving a great effect on the performance of the system. In addition to, the reliability ofthesystemwherethevehiclecanreacttosuddenfailureofoneormorevehiclesandcontinuefunctioningto achieve the main task [33]. Hungarian method was used in decentralized implementation in order to solve the OAP for the task assignment problem [29]. Auction algorithm was used in solving the task assignment forautonomous vehicles systems [35]. In [36], authors used genetic algorithm for solving the problem with decentralized approach.

V. OPTIMAL ASSIGNMENT

Over the past decade, many problems concerning the use of autonomous vehicles systems were investigated and discussed. One of the important problems aroused from the applications of autonomous vehicles systems in search and rescue missions and surveillance missions, is the "how to assign a set of tasks to a set of robots?". The optimal answer of this question is formulated to be a MRS problem called optimal task assignment.

1. Problem Definition and Characterization

Theoptimalassignmentproblem canbedefined asassigningaset ofneededtasks $T=\{T_1,T_2,\ldots T_m\}$,toa setof availablevehicles$V=\{V_1,V_2,\ldots V_n\}$.Theset$S$istheassignmentresult$S=\{(V_1,T_1),(V_2,T_2),\quad ,(V_k,T_k)\}$ $\forall\ 1 \leq k \leq m$. This assignment is then minimized or maximized according to a certain objective function or a set of objective functions. The increased demand on using autonomous vehicles systems in many complex problems increased the complexity of the optimal assignment problem. This complexity is due to many factors in the problem. These factors include the heterogeneous nature of used vehicles and needed tasks, the different constraints of the problem and the working environment [37].

2. Problem Modeling and Formulation

Task assignment must ensure that not only the search mission is achieved, but also the tasks are well distributed among the vehicles. The multi-vehicle task allocation problem here can be modeled and solved as an instance of the optimal assignment benchmark problem. The problem should be modeled first. The problem can bemodeled as a fully connected bi-partite graph as in Figure1.

The information collected by the vehicles from the surrounding environment about the victims' status and position is transferred to the base station. After that the base station processes the information and makes a decision which transfers to the vehicles in order to execute it. At this point, the problem can be put in the context of the optimal assignment problem. The optimal assignment problem is basically the "N agents- M jobs" problem where a single job can be assigned to only one agent in such a way that the overall cost of assignment is minimized [38].

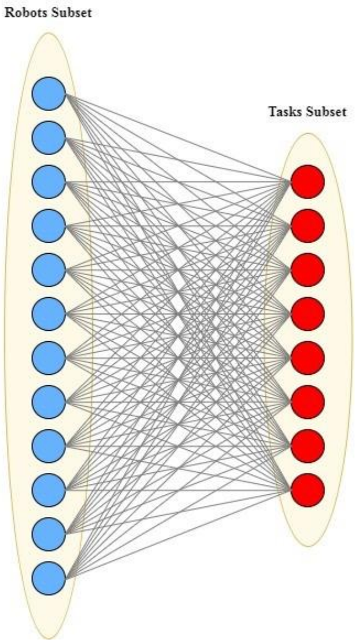

Figure 1: Fully-connected bipartite graph

As mentioned before the optimal assignment problem can be solved using different techniques and approaches. The objective function for evaluating the total cost of the assignment of M tasks to N vehicles presented in equation 1.

$$CostFunction = \sum_{i=1}^{N} \sum_{j=1}^{M} \alpha_{ij} (C1_{ij} + C2_{ij}) \qquad (1)$$

where,

$$C1_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \qquad (2)$$

and,

$$C2_{ij} = \begin{cases} 0, & \text{iff Resources of vehicle i matches needs of task j} \\ P, & \text{otherwise} \end{cases} \qquad (3)$$

Subjects to,

$$\sum_{j=1}^{M} \alpha_{ij} = 1 \;\; , 1 \leq i \leq n(4)$$

Equation 1 represents the objective function of the optimization problem which is the overall system utility. The cost function consists of two parts in equations 2and 3respectively. Where, equation 2 represents the Euclidean distance between the vehicle *i*and task *j*. Moreover, if the vehicle is loaded with other tasks the distance will be calculated as the whole distance transferred by the vehicle achieving its previously loaded tasks until it reaches the required task (P). In addition, equation 3 represents the penalty to vehicle *i*if its resources are not matching the needs task *j*. Moreover equation 4 represents the constraints of the function where each task is assigned only to one vehicle.

VI. PROPOSED APPROACH

In this section, the proposed approaches to solve the problem of context-aware optimal assignment in cooperaive autonomous vehicles are presented.

1. Proposed Approach: Overview

Figure 2 shows a block diagram for the assignment environment and its components. The environment consists of four components. First component is the assignment entity which is responsible for assigning the tasks to the available vehicles depending on their capabilities, status and position. Moreover the execution entities which take inputs from the assignment entity and execute the tasks. In our environment the execution entities are represented in the vehicles. After that, the tasks' execution is monitored for detection of any failures that could happen in the execution entities. Furthermore, in case of failures adaptation entity performs re-assignation for uncompleted tasks to other available vehicles, in order to ensure that all the needed tasks are achieved.

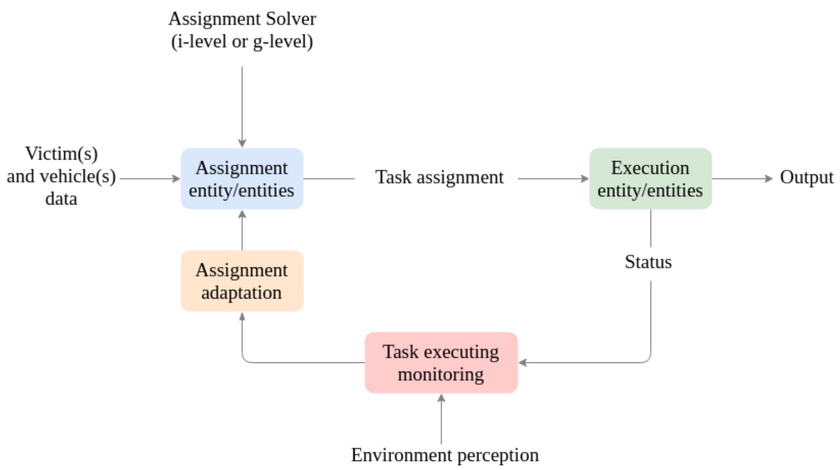

Figure 2: Overview of the proposed approach

2. Optimal Assignment

One of the fundamental issues that arises in search and rescue missions, which is the case study in this paper, is how to assign a set of tasks to a set of autonomous vehicles to effectively perform a given system level task. This task-vehicle assignment is called task allocation process or optimal task assignment process. This process may need to be continuously adjusted to adapt with the changes in the environment and/or group performance. This makes dynamic/adaptive task assignment one of the essential challenges for mobile autonomous vehicle systems. For a group of autonomous vehicles to effectively perform a search and rescue mission efficiently and reliably, they need to distribute the tasks of rescuing the detected victims amongst themselves. In this section the task assignment will be tackled in two aspects. The first aspect is to perform task assignment within autonomous vehicles system in the presence of global information shared between the vehicles and the base station. In contrast, the second aspect is to perform task assignment without sharing information globally, instead the vehicles can only share information within their neighborhood (communication range). The optimization problem of task assignment modeled and formulated in equation 1 will be solved using two approaches: stochastic and greedy approaches. The stochastic approach proposed is particle swarm optimization approach, while the deterministic greedy approach proposed is the Hungarian assignment method.

2.1. Task Assignment with Stochastic Solver

The stochastic solver used is particle swarm optimization. Particle swarm optimization is a population based stochastic optimization technique developed by Eberhart and Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling [39] [40]. Kennedy and Eberhart represented various approaches to mimic the dynamics of these social organisms. After that they reaches equations 5 and 6 to evaluate velocity and position respectively of each particle in the problem space:

$$V_{id}^{new} = WV_{id}^{old} + c1r1(P_{id} - X_{id}) + c2r2(P_{gd} - X_{id}) \tag{5}$$

$$X_{id}^{new} = V_{id}^{old} + V_{id}^{new} \tag{6}$$

Where, V_{id}^{new} represents the velocity of the particle which is the distance to be travelled by this particle in this iteration. V_{id}^{old} represents the velocity of the particle in the previous iteration. X_{id} represents the position of the particle in the problem space. Moreover, P_{id} is the best position achieved by this particle throughout the previous iterations. P_{gd} is the best position achieved in the problem space by all the particles in the population within the previous iterations. Furthermore, r_1 and r_2 are random numbers generated for each dimension d between [0,1]. W is the inertia weight of the particle which accommodates the fact that a bird (particle) cannot change its direction of movement suddenly. c_1 and c_2 are the balance factors for the cognitive component and social component respectively. They are used to control the effect of the historical values of the particle to evaluate its new values. Tuning the factors W ,c_1 and c_2 affects the motion of the particle in the problem space. Choosing large weight factor W forces the particles for global exploration of the problem space. In contrast small weight factor W forces the particle for exploitation of the current search area. Also choosing c_1 and c_2 in addition to W pro- vides a balance between global and local search. Particles' performance is evaluated with regards to its cost [41]. The iterations proceed until the termination criteria is met.

ThemappingbetweenarealcandidatesolutionandtheparticleinthePSOpopulationisimportanttoimplement a successful PSO algorithm [42]. A similar mapping between the solution and the particle resembles the onein[42] was used as in Figure 3. In this algorithm the particle was implemented as a vector of M dimensions equal tonumberoftasks.Eachcellcontainsthenumberofthevehicletowhichthistaskwithcellindexwasassigned for. Each dimension has limited possible values from the discrete set $S = \{P_i: 1 \leq i \leq N\}$; such that N isthenumber of vehicles in the system. Also, the particle contains a vector of velocities in all dimensions of the problemspacewhichequaltoM.Inadditiontoitcontainsthecostvalueoftheparticle(candidatesolution)ateach iteration.

The PSO population of the algorithm was implemented as a two-dimensional array of size (P,M), where P is the size of the PSO population. Each is a vector of dimension M which is number of tasks in the assignment problem. The algorithm begins with initializing a random solution with number of particles equal to population size. Each particle contain a complete assignment solution (candidate solution). The velocity of the initial solutionis set to zero. After that the cost of each particle is evaluated by cost function represented in equation 1. It then evaluates the global-best solution by comparing the cost value of the all the particles and choosing the least one (minimum assignment cost). Then it sets the local-best solution of each particle to the initial and only solution at this iteration.

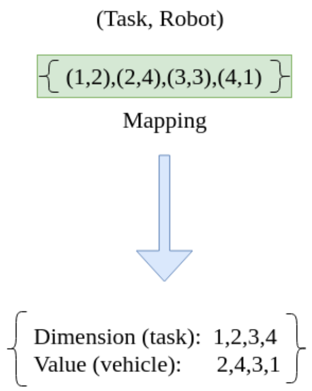

Figure 3: Mapping between candidate solution and PSO particle

2.2. Task Assignment with Deterministic Solver

To begin with, Hungarian algorithm must solve linear assignment problem where, the number of vehicles N equal to the number of tasks M. An imbalance in the number of vehicles and the number of tasks can be solved by adding dummy vehicles or tasks according to the problem [27]. Once the cost matrix is balanced, the Hungarian algorithm is ready to be applied on the cost matrix in order to solve the task allocation problem and gives the optimal assignment result.

The Hungarian optimization is implemented using the algorithm defined in [26] with a cost matrix of size (N,M). Original Hungarian technique has a drawback that it cannot solve an unbalanced assignment problem, where number of tasks is greater than number of vehicles, and to assign two or more tasks to one vehicle if it is necessary. In [43], authors presented a modified Hungarian approach for solving unbalanced assignment problem.

Their approach depends on formulating the cost matrix by putting the vehicles many times as needed to make the total number of vehicles more than the number of tasks. After that, the cost matrix could be balanced by adding dummy tasks to make it square matrix. Furthermore, the cost matrix is solved using normal Hungarian approach to assign tasks to vehicle.

3. Adaptation of Optimal Assignment

Increasingthesuccessrateoftaskachievementisoneoftheimportantdisciplinesintheresearchofoptimaltask assignment. Failed tasks affect the success rate in search and rescue missions which is our case study. Tasks could be uncompleted due to many reasons such as vehicles' failures. An adaptation entity is proposed as an approach to solve this problem. Adaptation entity processes information about failed vehicles in order to re-assign their tasks to available vehicles. The central coordinator mainly responsible for assigning tasks to vehicles after search stage is finished. In case of failure of the central coordinator and/or problems in its communication modules.Oneofthevehiclesbecomethecentralcoordinatorandtakestheroleinassigningtaskstoitselfand to the other vehicles. Despite of increasing the success rate of search and rescue missions and guaranteeing the achievement of nearly all discovered tasks, adaptation may increase the cost of totalassignment.

4. Global Information Scenario

This scenario resembles a black-board, where all the information can be seen and accessed by all the agents. To begin with, in this scenario, the vehicles are searching for victims in a disaster area. The search is applied by divide and conquer algorithm, where the search area is divided into designated areas and each vehicle is assigned an area of interest. The motion of a vehicle inside its assigned area follow a trajectory of square wave- form. When a vehicle find a victim, there is a task to be performed on it. The vehicle then sends data about the victim, its position and its status to the base station. Later, when the vehicles finishes searching for victims in the search area, comes the part of rescuing the discovered victims. The base station uses position and status of the victims found and the positions and the capabilities of the vehicles to make task assignment to each vehicle.

The task assignment is done by using particle swarm optimization or by Hungarian optimization. Once the task assignment process finishes, the base station sends an order to the vehicles to go to the victims' locations and perform certain actions. After that, the vehicle sends back an acknowledgement to the base station whether the operation is done or not. In case failure happens to one or more vehicles, it is reported to base station. The base station re-assign the uncompleted tasks to the available vehicles excluding the failed ones. The algorithm of global information scenario is presented in Algorithm 1.

Algorithm 1: Global information scenario algorithm

input : Area under disaster; Victims need to berescued;

output: Rescued victims;
initialization:
Multi-vehicle system;
Base station;
Communication network between the base station and the vehicles;
Divide the map into areas of interest equal to number of vehicles;
while *search stage is running* **do**
  each vehicle explores its own area ofinterest;
  if *victim is discovered***then**
  report to basestation;
  else
  vehicles complete exploring the area underdisaster;
Base station performs task assignment for victims' rescue;
Base station sends orders to each vehicle with its tasks;
while *rescue stage is running* **do**
  each vehicle tries to achieve its owntasks;
  if *vehicle is failed***then**
  Report to basestation;
  Base station re-assigns tasks of failed vehicle to other availablevehicles
  else
  vehicles complete performing theirtasks;
  if *all tasks are achieved***then**
  vehicles return to basestation;
  terminatemission.

5. Local Information Scenario

The scenario in which the vehicles do not have access to the global information is presented in this subsection. This could happen due to failures in the infrastructure. In such situation a communication relay is needed between vehicles to exchange information between the vehicles themselves or between the vehicles and the base station back and forth. In the study case presented in this paper (search and rescue), the vehicles do the search stage using divide and conquer algorithm. Furthermore, once a vehicle discovers a victim during exploration, it starts initiating an ad-hoc network to transfer the information about this victim to assignment unit.

Moreover, the unit in charge of the task assignment starts performing task allocation to vehicles. Finally, the task assignment is then transferred to the vehicles by the meaning of an ad-hoc network to start executing their assigned tasks. The unit in charge of performing task assignment process can be the base station in the centralized assignment or one of the vehicles in the distributed assignment.

5.1. Centralized Assignment

In the centralized architecture of the local information scenario, once a vehicle finds a victim during search process. It should report to the base station some information about this victim. This information contains position of the victim and its status (type of help needed). Due to lack of direct connection between the vehicle and the base station, it is required to initiate a network to pass the information to the base station. The vehicle starts a market-based approach for forming a network called auction approach using contract network protocol (CNP) as in Figure 4. This approach is described in the following stages:

- **Announcementstage**,thesendersetsalistofmessagesanddefinealltheirrequirementstoallvehicles.
- **Bidding stage**, only neighboring vehicles start bidding on the messagedelivery.
- **Selectionstage**,themessagesenderevaluatesallthebidsandstartthewinningdeterminationstrategy.
- **Assigning stage**, the message sender announces the winner by transmitting the message toit.

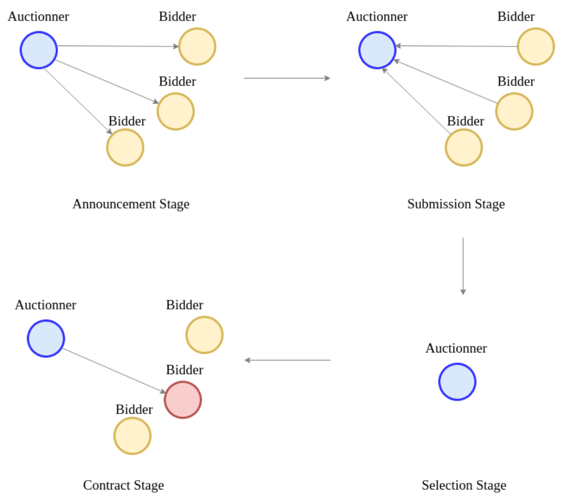

Figure 4: Contract network protocol stages

In the announcement stage, if the vehicle does not find any neighborhood vehicle in its communication range, it starts a random motion until it finds ones. Then it complete the auction process. This auction process is repeated many time until the message receiver becomes the base station. After certain time frame, the base station starts the task assignment process. Furthermore, the results transmitted to the vehicles in order to execute their tasks. This is also done using the auction process. Vehicles after finishing their search stage and passing information to the base station, each stays in its own assigned area. One of the vehicles lays in the communication range of the base station where, its assigned area is beside the base station. This vehicle takes the information of task assignment and then starts a random motion until it finds neighboring vehicles to auction on delivering the message.

Each vehicle receives the message of task assignment takes its task assignment part and repeats the auction process until another vehicle become the auctioneer. Then this vehicle goes to the location of its assigned task to execute it. Moreover, in case of vehicle failure it announces that to the vehicles in its neighborhood. The vehicle caught the information about the tasks of the failed vehicle begin an auction process on delivering that message. The auction process ends up when the message arrives to the base station. The base station re-assign the failed tasks to other vehicles. After that the vehicle reported the failure message to the base station takes the message of assignment. This vehicle starts random motion to deliver it to the vehicles. The local information scenario with centralized architecture follows the steps presented in Algorithm 2.

Algorithm 2: Local information scenario with centralized architecture algorithm

input : Area under disaster; Victims need to berescued;
output: Rescued victims;
1 **initialization:**
2 Multi-vehicle system;

Base station;
Divide the map into areas of interest equal to number of vehicles;
while *search stage is running* **do**
each vehicle explores its own area ofinterest;
if *victim is discovered***then**
vehicle starts an auction process on message delivery to base station;
else
vehicles complete exploring the area underdisaster;

Base station performs task assignment for victims' rescue;
Base station sends orders to the nearest vehicle in its communication range;
while *rescue stage is running* **do**
carrying message vehicle starts an auction process on delivering orders tovehicles;
if *a vehicle caught the message***then**
this vehicle starts auction process on delivering message tovehicles
else
the old vehicle tries to achieve its owntasks
if *vehicle is failed***then**
Reports failed tasks to the nearest neighboringvehicle;
Neighboring vehicle starts an auction to deliver message about failed tasks to the basestation;
Base station re-assigns tasks of failed vehicle to other availablevehicles;
Basestationtransmitsamessagecontainsre-assignmenttothenearestvehicleinits communicationrange;
else
vehicles complete performing theirtasks;
if *all tasks are achieved***then**
vehicles return to basestation;
terminatemission.

5.2. Distributed Assignment

In this case there is no base station or central coordinator for the vehicles. Instead the role of the central unit is done using the vehicles themselves. as shown in Algorithm 3. In contrast with the centralized architecture, the auction method used here for direct task assignment not for message delivery. Once a vehicle discovers a victim, it starts moving in a random motion searching for a neighborhood vehicles. Moreover, the vehicle holding the tasks sends an auction request to the neighborhood vehicles holding the requirements for the each task in the announcement stage of the auction process. Then, the neighborhood vehicles send their bids on the available tasks. These bids can be single-item bids or combinatorial bids for a group of tasks. The vehicle holding the tasks evaluates the bids of the neighborhood vehicles by a certain strategy in evaluation stage.

The strategy used here is the distance from the vehicle to the required task as well as matching the capabilities of the vehicles to the task needs. Finally the vehicle holding the tasks announces the winners by transmitting the tasks to them. After that, the vehicle continues its normal motion until it finishes the search stage or discovers another victim to start the auction process again. In vehicle's failure cases, the failed vehicle announce to the vehicles in its neighborhood its state and the uncompleted tasks. The vehicle which catch the message from the failed vehicle start an auction process on the uncompleted tasks to re-assign them to other available vehicles.

Algorithm 3: Local information scenario with centralized distributed algorithm

input : Area under disaster; Victims need to berescued;
output: Rescued victims;
initialization:
Multi-vehicle system;
Divide the map into areas of interest equal to number of vehicles;
while *search stage is running* **do**
each vehicle explores its own area ofinterest;
if *victim is discovered***then**
vehiclestartsanauctionprocessonassigningdiscoveredvictim'stasktoneighboringvehicles;
if *a vehicle is assigned the task***then**
the old vehicle completes exploring its area ofinterest;
the assigned vehicle goes to achieve itstask;

after achieving the task it returns to complete exploring its area ofinterest;
else
vehicle makes random motion in the map auctioning on the taskassignment;
else
vehicles complete exploring the area underdisaster;
if *all tasks are achieved the map is explored***then**
vehicles return to basestation;
terminatemission.

VII. EXPERIMENTAL STUDY

1. Experimental Setup

A test environment is implemented robot operating system (ROS) and simulated in Gazebo 3D-simulator. The machine used to run the simulation was a computer (Core i7-3632QM with 2.2 Ghz speed, 1.00 GB GPU, 4.00 GB RAM) running Linux-based operating system. It was planned to use a large number of vehicle in simulations starting with 25 vehicles, but due to lack of High Performance Computing (HPC) facility, only four vehicles used.

2. Evaluation Metrics

The following evaluation metrics are considered: computational cost (taking into consideration the computational run time of the algorithm) and task allocation cost. The results are divided into three parts; the first part is related to the results of the global information scenario, the second part is related to results of the local information scenario with centralized architecture and the third part is related to the local information scenario with distributed architecture. For the first and second parts, results of three experiments are presented. The first experiment tests the normal approach of task assignment with PSO and Hungarian optimization techniques. The second experiment tests the adaptation in case of vehicle's failure with PSO and Hungarian optimization techniques. The third experiment tests also the adaptation in case of another vehicle's failure with PSO and Hungarian optimization techniques. For the third part of the results, two experiments are presented. The first experiment tests the normal task assignment using auction method, the second experiment tests the adaptation in case of vehicle's failure with auction method.

3. Experimental Scenarios

Three simulation scenarios were implemented for testing the proposed approaches. The scenarios aim to explore the points of strength and points of weakness in order to perform a good communication between cooperative vehicles. These scenarios are global information scenario and local information scenario which is implemented in two forms, centralized and distributed. After implementing these scenarios, a comparison was made for validating their performance according to the evaluation matrices mentioned and explained in the previous section.

The vehicles are performing search and rescue mission in a disaster situation. The location of the disaster will be the search area for the vehicles in their missions. The vehicles used in this scenario are ground autonomous vehicles of type Husky. Husky mobile vehicles is created by Clearpath Robotics [44]. Each vehicle isequipped withlaserfinderandacamera,(asinFigure5).Thetypeofenvironmentusedinsimulationsisagroundrough terrain, (as in Figure 5), which resembles the disasters'areas.

Figure 5: Autonomous ground vehicles used in experimental study

The vehicles in the system have three different types: search vehicles which are capable of searching, detecting the victims and communicating with other vehicles as well as the base station, rescue I vehicles which in addiion to perception capabilities, they can move obstacles away with the gripper and rescue II vehicles which carry some tools that can be used for injuries in addition to perception capabilities. The system in these experiments consists of four Husky vehicles, two of type rescue II vehicles, one Search vehicle and one rescue I vehicle.Three types of victims are considered in the experiments. Type-I represents victims who do not suffer from bad injuries according to the information coming from the search/reconnaissance vehicles. Type-II represents victims with bad injuries and are in need for an immediate medical help. Type-III represents trapped or concealed victims who need a rescue vehicle equipped with gripper and able to get them out from their current trap. The three proposed scenarios are:

3.1. Global Information Scenario

Figure 6 represents collected figures showing the motion of the four vehicles in the global scenario simulation. A video for the simulation is presented here. The victims' discovery is shown in Table 1.

Table 1
Victims discovery stage results

	Victim type		Vehicle type
Victim 1	I	Vehicle 4	Rescue II
Victim 2	II	Vehicle 2	Rescue II
Victim 3	III	Vehicle 2	Rescue II
Victim 4	II	Vehicle 3	Search

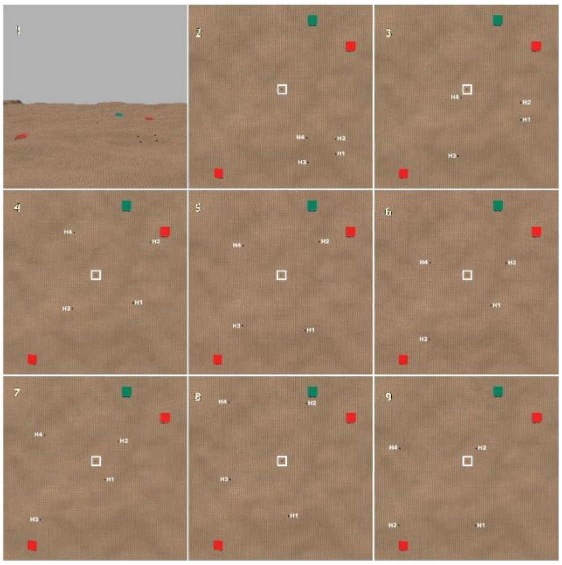

Figure 6: Motion of four vehicles during search in global scenario

Aftersearchingforthevictimsinthewholesearchareabythevehicles,fourvictimswerediscovered.Thesevictims are: one of type I, two of type II and one of type III. Then, the base station have all the information about the victims, their position and status. Also, the information about the vehicles is known prior to the base station. The base station uses this information for formulating and solving task assignment problem for assigning the tasks of the four victims to the vehicles such that every victims gets the help needed. An experiment was simulated with the data shown in tables 2 and 3 for the vehicles and the victimsrespectively.

Table 2: Data of vehicles after search stage

	Pos.n in X-direction	Pos. in Y-direction	Capabilities
Veh 1	11.99	35.06	Rescue I
Veh 2	57.95	34.15	Rescue II
Veh 3	11.71	78.99	Search
Veh 4	57.73	79.98	Rescue II

The particle swarm optimization was implemented with the following values for its parameters:
1. The size of the population equals to the number of tasks(victims).
2. The weight factor in equation 5 is0.9.
3. The cognitive part factor c1 and social part factor c2 are set both equal to1.0.
4. Stopping criteria is reaching maximum iteration equal to 100iterations

Table 3: Data of discovered victims

	Pos. in X-direction	Pos. in Y-direction	Type
Vic 1	49.27	46.56	III
Vic 2	80.48	5.59	II
Vic 3	95.78	26.52	I
Vic 4	10.93	86.34	II

The PSO algorithm ran five times. The results are listed in Table 4 and the best solution is highlighted. It took the PSO algorithm 0.007 seconds to reach the optimal solution in the best case with total cost of 223.815 distance units.

Table 4: Experiment 1: Task assignment using PSO-based algorithm

Vic 1	Vic 2	Vic 3	Vic 4	Cost	Comp. time (Sec.)
Veh 1	**Veh 2**	**Veh 3**	**Veh 4**	**223.8**	**0.007**
Veh 1	Veh 4	Veh 2	Veh 2	395.3	0.011
Veh 4	Veh 4	Veh 3	Veh 2	438.7	0.031
Veh 4	Vehe 4	Veh 2	Veh 2	545.9	0.007
Veh 1	Veh 2	Veh 1	Veh 2	353.2	0.008
					Normalized time 0.0128

For purposes of testing adaptation vehicle 4 is subjected to failure. This failure was reported to the base station in order to re-assign its task (victim 4) to another vehicle. The PSO algorithm ran five times. The results are listed in table 5 and the best solution is highlighted. It took the PSO algorithm 0.012 seconds to reach the optimal solution in the best case with total cost of 212.429 distance units.

Table 5: Experiment 2: Task assignment using PSO-based algorithm after failure of vehicle 4

Vic 1	Vic 2	Vic 3	Vic 4	Cost	Comp. time (Sec.)
Veh 1	**Veh 2**	**Veh 3**	**Veh 2**	**212.4**	**0.012**
Veh 1	Veh 2	Veh 2	Veh 2	376.7	0.013
Veh 2	Veh 2	Veh 3	Veh 1	579.8	0.007
Veh 1	Veh 2	Veh 2	Veh 2	376.7	0.009
Veh 2	Veh 2	Veh 3	Veh 2	394.4	0.033
					Normalized time 0.0148

Anotheradaptationtestwasconductedwherevehicle3failed.ThePSOalgorithmranfivetimes.Theresultsare listed in table 6 and the best solution is highlighted. It took the PSO algorithm 0.01 seconds to reach the optimal solution in the best case with total cost of 361.663 distanceunits.

Table 6: Experiment 3: Task assignment using PSO-based algorithm after failure of vehicle 3

Vic 1	Vic 2	Vic 3	Vic 4	cost	Comp. time (Sec.)
Veh 1	**Veh 2**	**Veh 2**	**Veh 4**	**361.6**	**0.01**
Veh 1	Veh 2	Veh 1	Veh 2	353.2	0.01
Veh 4	Veh 2	Veh 4	Veh 2	542.4	0.008

Veh 4	Veh 4	Veh 2	Veh 2	545.9	0.008
Veh 1	Veh 4	Veh 2	Veh 2	395.3	0.007
					Normalized time 0.0086

The second algorithm used is Hungarian-based optimization algorithm. The Hungarian algorithm was tested withthesamedatamentionedin Tables 2and3forthevehiclesandthevictimsrespectivelyasforexperiment 1.Itresultsinassignmentas(victim1→vehicle1,victim2→vehicle2,victim3→vehicle3andvictim4→ vehicle 4) with total cost of 223.158 distance units in 0.105 milliseconds.

Moreover, an experiment is conducted for testing adaptation with same conditions as experiment 2, where vehicle4issubjectedtofailure.Theresultsare(victim1→vehicle1,victim2→vehicle2,victim3→vehicle3 andvictim4→vehicle2)withtotalcostof211.161distanceunitsin0.468milliseconds.

Another experiment was conducted with the same conditions as experiment 3 (vehicle 3 is the one subjected to failure).Itresultsinassignmentas(victim1→vehicle1,victim2→vehicle2,victim3→vehicle2andvictim 4→vehicle4)withtotalcostof361.548distanceunitsin0.392milliseconds.

3.2. Central Local InformationScenario

Figure7showsthemotionofthevehiclesduringthesimulation.Avideoforthesimulationispresented*here*. Table 7showtheresultsofsearchstage.Positionsandtypesofthediscoveredthesearchvehiclesaftersearch stage shown in Table 8. While positions and types of discovered victims are shown in Table9.

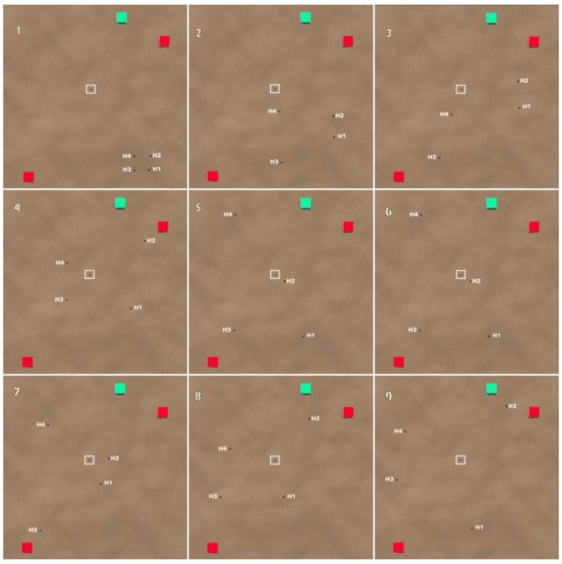

Figure 7: Motion of four vehicles during search in local information scenario with centralized allocator

Table 7: Victims discovery after stage results

	Victim type		Vehicle type
Victim 1	I	Vehicle 4	Rescue 2
Victim 2	II	Vehicle 2	Rescue 2
Victim 3	III	Vehicle 2	Rescue 2
Victim 4	II	Vehicle 3	Search

Table 8 Data of vehicles after search stage

	Pos in X-direction	Pos in Y-direction	Capabilities
Veh 1	26.86	47.79	Rescue I
Veh 2	60.43	38.14	Rescue II
Veh 3	9.41	88.28	Search
Veh 4	61.15	89.66	Rescue II

Table 9: Data of discovered victims

	Pos in X-direction	Posn in Y-direction	Type
Vic 1	51.34	45.77	III
Vic 2	79.87	4.98	II
Vic 3	94.99	26.54	I
Vic 4	9.79	86.79	II

The PSO algorithm ran five times for ensuring best results are obtained. The results generated are listed in table 10 and the best results is highlighted. It took 0.008 seconds for the PSO algorithm to reach optimal assignment with total cost of 206.704 distance units.

Table 10: Experiment 1: Task assignment using PSO-based algorithm

Vic 1	Vic 2	Vic 3	Vic 4	Cost	Comp time (Sec.)
Veh 1	**Veh 2**	**Veh 3**	**Veh 2**	**206.7**	**0.008**
Veh 1	Veh 4	Veh 1	Veh 2	493.0	0.006
Veh 1	Veh 1	Veh 3	Veh 4	531.1	0.016
Veh 1	Veh 2	Veh 3	Veh 1	391.1	0.033
Veh 1	Veh 4	Veh 4	Veh 2	463.7	0.022
					Normalized time
					0.017

Vic 1	Vic 2	Vic 3	Vic 4	cost	Comp time (Sec.)
Veh 1	**Veh 2**	**Veh 1**	**Veh 2**	**323.536**	**0.009**
Veh 1	Veh 2	Veh 2	Veh 2	365.414	0.016
Veh 1	**Veh 2**	**Veh 1**	**Veh 2**	**323.429**	**0.009**
Veh 1	Veh 2	Veh 4	Veh 2	372.496	0.008
Veh 1	Veh 2	Veh 4	Veh 2	372.496	0.008
					Normalized time
					0.01

For testing the adaptation vehicle 3 is subjected to failure in experiment 2. This was reported to the base station by vehicle 1. The base station in turn re-assign the tasks of vehicle 3 (victim 3) to another available vehicles after excluding vehicle 3. The PSO algorithm ran five times to get several solution. The results are listed in table 11 and the best solutions are highlighted. It took PSO 0.009 seconds to reach a sub-optimal assignment of cost 323.429.

Table 11: Experiment 2: Task assignment using PSO-based algorithm after failure of vehicle 3

Vic 1	Vic 2	Vic 3	Vic 4	cost	Comp time (Sec.)
Veh 1	**Veh 2**	**Veh 1**	**Veh 2**	**323.536**	**0.009**
Veh 1	Veh 2	Veh 2	Veh 2	365.414	0.016
Veh 1	**Veh 2**	**Veh 1**	**Veh 2**	**323.429**	**0.009**
Veh 1	Veh 2	Veh 4	Veh 2	372.496	0.008
Veh 1	Veh 2	Veh 4	Veh 2	372.496	0.008
					Normalized time
					0.01

Another test is conducted by re-assigning tasks of vehicle 2 after it is subjected to failure in experiment 3. The PSO algorithm ran five times to obtain several results. The results generated are listed in table 12 and the best solution is highlighted. It took PSO 0.009 seconds to reach sub-optimal assignment with total cost of 303.213 distance units.

Table 12: Experiment 3: Task assignment using PSO-based algorithm after failure of vehicle 2

Vic 1	Vic 2	Vic 3	Vic 4	Cost	Comp. time (Sec.)
Veh 1	**Veh 4**	**Veh 3**	**Veh 4**	**303.2**	**0.009**
Veh 1	Veh 1	Veh 3	Veh 4	403.4	0.007
Veh 1	Veh 4	Veh 3	Veh 1	439.4	0.009
Veh 1	Veh 1	Veh 3	Veh 1	627.2	0.006

Veh 1	Veh 1	Veh 3	Veh 4	403.4	0.007
					Normalized time 0.0076

Hungarian-based optimization algorithm used to solve the assignment problem. The algorithm used withsame data mentioned in Tables 8 and 9 respectively and same conditions as experiment 1. The results of assignment are (victim1→vehicle1,victim2→vehicle2,victim3→vehicle3andvictim4→vehicle4). Ittook0.118 milliseconds to reach an assignment of total cost 207.534 distanceunits.

For purposes of testing adaptation, experiment 2 is conducted where, vehicle 3 is subjected to failure. The tasks arere-assignedbasedonavailablevehicleswhicharevehicles1,2and4.Theresultsare(victim1→vehicle1, victim2→vehicle2,victim3→vehicle4andvictim4→vehicle4).Thealgorithmreachesassignmentwith total cost of 349.18 distance units in 0.471 milliseconds.

Experiment3isconducted,butthistimevehicle2issubjectedtofailure,theresultsare(victim1→vehicle1, victim2→vehicle3,victim3→vehicle4andvictim4→vehicle4).Thetotalassignmentcosts246.116distance units and it is generated in 0.524 milliseconds.

3.3. Distributed Local Information Scenario

	Victim type		Vehicle type
Victim 1	I	Vehicle 4	Rescue 2
Victim 2	II	Vehicle 2	Rescue 2
Victim 3	III	Vehicle 2	Rescue 2
Victim 4	II	Vehicle 3	Search

In this scenario, figure 8 shows the motion of vehicles. A video for the simulation is presented here. The vehicles discovered the victims as shownintable13. The results of the assignment processare (victim1→vehicle1, victim2→vehicle3,victim3→vehicle4andvictim4→vehicle2)

Table 13: Victim's discovery stage results

	Victim type		Vehicle type
Victim 1	I	Vehicle 4	Rescue 2
Victim 2	II	Vehicle 2	Rescue 2
Victim 3	III	Vehicle 2	Rescue 2
Victim 4	II	Vehicle 3	Search

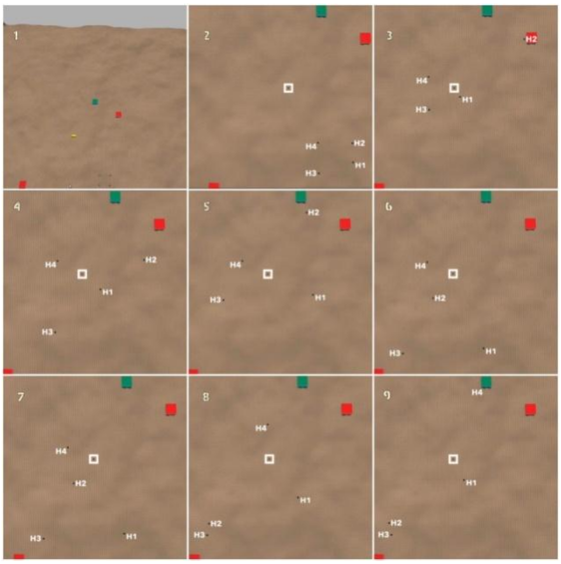

Figure 8: Motion of four vehicles during search in local information scenario with distributed allocator

Another experiment is conducted for testing the adaptation approach. Victims are discovered by vehicles as shown in table 14. After each vehicle discover a victim the auction process is used for victims' task assignment. Theresultsoftaskassignmentare(victim1→vehicle1,victim2→vehicle2,victim3→vehicle4andvictim 4→vehicle3).

Table 14: Victims discovery stage results

	Victim type		Vehicle type
Victim 1	I	Vehicle 4	Rescue 2
Victim 2	II	Vehicle 2	Rescue 2
Victim 3	III	Vehicle 2	Rescue 2
Victim 4	II	Vehicle 3	Search

Furthermore, vehicle 2 is subjected to failure. The vehicle reported its failure to the nearest neighbor vehicle which is vehicle 4. Moreover, vehicle 4 starts an auction process on assignment the tasks of vehicle 2 (victim 2). The auction process ends with announcement of the winning bidder. The winner is vehicle 3 which is then responsible for achieving victim 2 task. The results of task assignment after adaptation process of vehicle 2 failurearepresentedas(victim1→vehicle1,victim2→vehicle3,victim3→vehicle4andvictim4→vehicle3).

VIII. CONCLUSIONS

The contributions in this research is mainly about implementation a realistic scenarios for search and rescue missions in disaster areas using cooperative autonomous vehicles in Robot Operating System (ROS). Moreover, these scenarios investigate the problem of optimal task assignment under different communication capabilities and different organizational architectures. Finally, a comparison between stochastic and deterministic optimization techniques was conducted. The analysis of experimental results presented in this paper showed that the greedy algorithms like Hungarian assignment method is better than stochastic techniques in terms of computational time in small-scale problems. While both particle swarm optimization and Hungarian-based optimization showed the same performance in terms of reaching optimal or sub-optimal assignment cost. The stochastic techniques is better suited for large-scale complex problems. The lack of high performance computing facilities con- strained the testing of large-scale cases with complex data. Tables 15 and 16 show the results of the two solvers used in the global information and local information scenarios respectively.

Table 15: Comparative table for global information scenario

Particle Swarm optimization		
	Cost	Time (ms)
Without failure	**223.815**	7
With failure (Exp.I)	**212.429**	12
With failure (Exp.II)	361.663	13
Hungarian-based optimization		
	Cost	Time (ms)
Without failure	292.997	**0.2**
With failure (Exp.I)	223.158	**0.1**
With failure (Exp.II)	**211.161**	**0.4**

Table 16: Comparative table for local information scenario

Particle Swarm optimization		
	Cost	Time (ms)
Without failure	**206.704**	8
With failure (Exp.I)	**323.536**	9
With failure (Exp.II)	303.213	9
Hungarian-based optimization		
	Cost	Time (ms)
Without failure	207.534	**0.1**
With failure (Exp.I)	349.188	**0.4**
With failure (Exp.II)	**246.116**	**0.5**

As future work, different types of tasks as time-extended, tight tasks that cannot be done independently and tasks that can be performed in sequence, in branch or jointly will be studied. A probabilistic victim generator instead of the deterministic one will be used and High performance computing (HPC) facility will be used to testaswarmofvehiclesandtheefficacyofstochasticoptimizers.Moreover,thescalabilityofthesystemcanbe testedinthefuturewiththreedifferentdimensions:thesizeofasystemwithrespecttothenumberofinvolved tasks, the geographical size and the manageability of the system as the number of interconnected robots increase.

REFERENCES

[1]. C.WannousandG.Velasquez,"UnitedNationsofficefordisasterriskreduction(unisdr)—unisdr'scontri- butiontoscienceandtechnologyfordisasterriskreductionandtheroleoftheinternationalconsortiumon landslides (icl)," in *Workshop on World Landslide Forum*. Springer, 2017, pp.109–115.

[2]. B.ShahandH.Choset,"Surveyonurbansearchandrescuerobots,"*JournaloftheRoboticsSocietyof Japan*, vol. 22, no. 5, pp. 582–586,2004.

[3]. "A. khamis, "role of cooperation in multi-robot systems", keynote speech at international conference on industryacademiacollaboration(iac2014),doi10.13140/rg.2.1.2910.6720,https://goo.gl/k7a7zu,date= October, 2017, note = Accessed:2018-08-20."

[4]. T.Arai,E.Pagello,andL.E.Parker,"Advancesinmulti-robotsystems,"*IEEETransactionsonrobotics and automation*, vol. 18, no. 5, pp. 655–661,2002.

[5]. W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in *RoboticsandAutomation,2000.Proceedings.ICRA'00.IEEEInternationalConferenceon*,vol.1.IEEE, 2000, pp.476–481.

[6]. A. R. Mosteo and L. Montano, "Simulated annealing for multi-robot hierarchical task allocation with flexibleconstraintsandobjectivefunctions,"in*WorkshoponNetworkRobotSystems:Toward Intelligent Robotic Systems Integrated with Environments". IROS*,2006.

[7]. Z.Jian,P.Zhihong,andL.Bo,"Multi-taskallocationofucavsconsideringtimecostandhardtimewindow constraints," in *Control Conference (CCC), 2012 31st Chinese*. IEEE, 2012, pp.2448–2452.

[8]. P.Oberlin,S.Rathinam,andS.Darbha,"Today'stravelingsalesmanproblem,"*IEEErobotics&automa- tion magazine*, vol. 17, no. 4, pp. 70–77,2010.

[9]. S. Sariel-Talay, T. R. Balch, and N. Erdogan, "Multiple traveling robot problem: A solution based on dy- namictaskselectionandrobustexecution,"*IEEE/ASMETransactionsOnMechatronics*,vol.14,no.2,pp. 198–206,2009.

[10]. U. Endriss, "Lecture notes on fair division,"2010.

[11]. T. Walsh, "Online cake cutting," *Algorithmic Decision Theory*, pp. 292–305,2011.

[12]. I.Kash,A.D.Procaccia,andN.Shah,"No agent left behind: Dynamic fair division of multipler esources," *Journal of Artificial Intelligence Research*, vol. 51, pp. 579–603, 2014.

[13]. R.ZivanandE.Segev,"Trustbasedefficiencyforcakecuttingalgorithms,"*TRUSTINAGENTSOCIETIES (TRUST-2011)*, p. 141,2011.

[14]. A.SahuandR.Tapadar, "Solvingtheassignmentproblemusinggeneticalgorithmandsimulatedanneal- ing." in *IMECS*, 2006, pp.762–765.

[15]. B.KaleciandO.Parlaktuna,"Market-basedmulti-robottaskallocationusingenergy-basedbidcalcula- tions," *International Journal of Robotics and Automation*, vol. 27, no. 4, p. 396,2012.

[16]. C. Liu and A. Kroll, "A centralized multi-robot task allocation for industrial plant inspection by using a* andgeneticalgorithms,"in*Artificialintelligenceandsoftcomputing*. Springer Berlin/Heidelberg, 2012, pp.466–474.

[17]. B. Kalecı, O. Parlaktuna, M. Özkan, and G. Kirlik, "Market-based task allocation by using assignment problem,"in*SystemsManandCybernetics(SMC),2010IEEEInternationalConferenceon*. IEEE, 2010, pp.135–141.

[18]. N.AtayandB.Bayazit,"Mixed-integerlinearprogrammingsolutiontomulti-robottaskallocationprob- lem,"2006.

[19]. P.Sujit,J.George,andR.Beard,"Multipleuavtaskallocationusingparticleswarmoptimization,"*AIAA Paper*, no. 2008-6837,2008.

[20]. R.Poli,"Analysisofthepublicationsontheapplicationsofparticleswarmoptimisation,"*JournalofArtifi- cial Evolution and Applications*, vol. 2008,2008.

[21]. X. Li and H.-x. Ma, "Particle swarm optimization based multi-robot task allocation using wireless sensor network," in *Information and Automation, 2008. ICIA 2008. International Conference on*.IEEE, 2008, pp. 1300–1303.

[22]. W. Li and W. Zhang, "Method of tasks allocation of multi-uavs based on particles swarm optimization[j]," *Control and Decision*, vol. 9, p. 017, 2010.

[23]. S.-Y. Ho, H.-S. Lin, W.-H. Liauh, and S.-J. Ho, "Opso: Orthogonal particle swarm optimization and its

applicationtotaskassignmentproblems,”*IEEETransactionsonSystems,Man,andCybernetics-PartA: Systems and Humans*, vol. 38, no. 2, pp. 288–298,2008.
[24]. P.-Y. Yin, S.-S. Yu, P.-P. Wang, and Y.-T. Wang, “A hybrid particle swarm optimization algorithm for optimaltaskassignmentindistributedsystems,”*ComputerStandards&Interfaces*,vol.28,no.4,pp.441–450, 2006.
[25]. H.W.Kuhn,“Thehungarianmethodfortheassignmentproblem,”*NavalResearchLogistics(NRL)*,vol.2, no. 1-2, pp. 83–97,1955.
[26]. H.W.Kuhn,“Variantsofthehungarianmethodforassignmentproblems,”*NavalResearchLogistics (NRL)*, vol. 3, no. 4, pp. 253–258,1956.
[27]. G.A.Korsah,A.Stentz,andM.B.Dias,“Acomprehensivetaxonomyformulti-robottaskallocation,”*The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1495–1512,2013.
[28]. N.Michael,M.M.Zavlanos,V.Kumar,andG.J.Pappas,“Distributedmulti-robottaskassignmentandfor- mation control,” in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*.IEEE, 2008, pp.128–133.
[29]. S. Giordani, M. Lujak, and F. Martinelli, “A distributed algorithm for the multi-robot task allocation problem,”in*InternationalConferenceonIndustrial,EngineeringandOtherApplicationsofAppliedIntelligent Systems*. Springer, 2010, pp.721–730.
[30]. J. David, R. Valencia, and K. Iagnemma, “Task assignment and trajectory planning in dynamic environmentsformultiplevehicles,”in*RSS2016WorkshoponTaskandMotionPlanning,Ann Arbor, Michigan, USA, June 19, 2016*,2016.
[31]. M.B.Dias,R.Zlot,N.Kalra,andA.Stentz,“Market-basedmultirobotcoordination:Asurveyandanaly- sis,” *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1257–1270,2006.
[32]. K.Al-YafiandH.Lee,“Centralizedversusmarket-basedapproachestomobiletaskallocationproblem: State-of-the-art,”2009.
[33]. A.El-mogy,“Market-basedframeworkformobilesurveillancesystems,”Ph.D.dissertation,Universityof Waterloo,2010.
[34]. C. Liu and A. Kroll, “A centralized multi-robot task allocation for industrial plant inspection by usinga* and genetic algorithms,” *Springer-VerlagBerlin Heidelberg*,2012.
[35]. H.-L. Choi, L. Brunet, and J. P. How, “Consensus-based decentralized auctions for robust taskallocation,” *IEEE transactions on robotics*, vol. 25, no. 4, pp. 912–926, 2009.
[36]. G. Ping-An, C. Zi-Xing, and Y. Ling-Li, “Evolutionary computation approach to decentralized multi-robot task allocation,” in *Natural Computation, 2009. ICNC’09. Fifth International Conference on*, vol. 5.IEEE, 2009, pp.415–419.
[37]. M.Badreldin,A.Hussein,andA.Khamis,“Acomparativestudybetweenoptimizationandmarket-based approachestomulti-robottaskallocation,”*AdvancesinArtificialIntelligence*,vol.2013,p.12,2013.
[38]. B.P.GerkeyandM.J.Matarić,“Aformalanalysisandtaxonomyoftaskallocationinmulti-robotsys- tems,” *The International Journal of Robotics Research*, vol. 23, no. 9, pp. 939–954,2004.
[39]. R.EberhartandJ. Kennedy,“Anewoptimizerusingparticleswarmtheory,”in*MicroMachineandHuman Science, 1995. MHS ’95., Proceedings of the Sixth International Symposium on*,1995.
[40]. R. C. Eberhart, Y. Shi, and J. Kennedy, “Swarm intelligence (the morgankaufmann series inevolutionary computation),”2001.
[41]. Y.ShiandR.C.Eberhart,“Empiricalstudyofparticleswarmoptimization,”in*EvolutionaryComputation, 1999. CEC 99. Proceedings of the 1999 Congress on*, vol. 3. IEEE, 1999, pp.1945–1950.
[42]. A.Salman,I.Ahmad,andS.Al-Madani,“Particleswarmoptimizationfortaskassignmentproblem,”*Microprocessors and Microsystems*, vol. 26, no. 8, pp. 363–371,2002.
[43]. N. Betts and F. J. Vasko, “Solving the unbalanced assignment problem: Simpler is better,” *American Journal of Operations Research*, vol. 6, no. 04, p. 296,2016.
[44]. “Husky ugv, https://www.clearpathrobotics.com/husky-unmanned-ground-vehicle-robot/, note =Accessed: 2018-06-30.”

IOSR Journal of Engineering (IOSRJEN) is UGC approved Journal with Sl. No. 3240, Journal no. 48995.

Abdelrahman M. Ezz. “Adaptive Optimal Task Assignment for CooperativeAutonomous Vehicles.” IOSR Journal of Engineering (IOSRJEN), vol. 09, no. 07, 2019, pp. 43-60