Multi-Robot Coordination in Transport Mission

Kristel Verbiest[1], Eric Colon[2]
[1, 2] Royal Military Academy – Department of Mechanical Engineering
Av. de la Renaissance 30 – 1000 Brussels
email: kristel.verbiest@rma.ac.be, eric.colon@rma.ac.be

Abstract: **In dangerous tasks it can be advantageous to deploy robots to keep human participation to a minimum. Especially a team of cooperating robots can provide a valuable asset, as is the case in the transport problem studied here. The goal is to transport goods from one location to another through unknown hostile terrain as fast and safe as possible with the aid of a robot team. A convoy of larger robots carries the actual goods, while smaller robots assist in gathering information of the direct surroundings of the convoy and provide protection. Global path planning is used to move the convoy optimally, with current information available, complying with several constraints on the way. The small robots act in a coordinated way to maximize their information gain by the use of local planning. They provide new map information and alert the presence of adversaries. During the mission the global plan is continuously updated to account for this new information.**

Keywords: **multi-robot coordination, path planning, fuzzy logic control, formation control**

I. INTRODUCTION

In order to keep humans safe, robots are deployed more and more to execute dangerous tasks. For some more complicated tasks the use of a multi-robot system can be advantageous, as there are some tasks that are not achievable by a single robot. Moreover they will perform the task faster and more efficiently. Several dangerous scenarios are possible where the use of a robot team could be beneficial, as is the case in the transport mission studied here. The objective of the mission is the transport of goods from one location to another in a hostile unknown terrain. A team of cooperating robots aid in performing this task as generally they will accomplish a task faster and better than individual robots. The robots are required to complete their missions without collisions, preferably as fast and safe as possible. The robot team consists out of a convoy of larger robots that carry the actual goods and some smaller scout robots that assist in gathering information of the direct surroundings of the convoy and provide protection.

One approach could be to coordinate the movement of all the robots centrally to move them optimally to perform their task. However for large robot teams this approach will be intractable as the computations will increase exponentially. Another downside of this approach is that with the failure of the central unit all robots will be lost.

On the other hand, in a distributed approach every robot will act largely independently, acting on information that is locally available through its sensors. A robot may coordinate with other robots in its vicinity to accomplish a task it cannot accomplish by itself. This approach will require less computation and communication. The disadvantage is that the task can be accomplished much less efficiently than in the centralized approach.

To accomplish the transport task described here, a type of hybrid structure between the centralized and distributed approach is used. The approach is centralized in the sense that the scouts forward all their sensor reading towards the convoy. The convoy uses all this information to make a decision for itself. It is distributed in the sense that the scouts act only on their sensor readings and the robots in their vicinity. The desired group behavior for the scouts will be to move in such a way that will optimize their information gain, while not straying too far from the convoy to ensure its safety.

A simulator was developed in C++ to evaluate the strategy.

II. RELATED WORK

In the field of robotics the focus is shifting more and more from single-robot towards multi-robot systems. Previous work in multi-robot coordination is primarily done in the field of search and rescue, exploration, surveillance… Different approaches exist in controlling a robot team. A classification of different multi-robot systems and task allocation in multi-robot systems is given in [1] and [2] respectively. The concept of using a market-based model to allocate tasks between robots has been used in numerous applications, an overview in given in [3].

The approach taken here leans toward a leader-follower [4] task, where the scouts follow the leader, the convoy. In the transport task an added constraint is that the scouts must maximize their information gain while following the convoy. In doing so, they abide by some simple rules resulting them to ‘flock’ together in a circular formation. The circular formation is emergent behavior arising from simple rules that are followed by the scout robots, without central coordination. The multi-robot system here is emergent as opposed to intentional where the robots cooperate explicitly by negotiation or allocation of tasks. Other approaches toward formation control in multi-agent systems can be found in following review [5].

To ensure smooth movement of the scout robots a fuzzy logic controller as inspired by [6] and [7] is utilized to control robot navigation.

III. TRANSPORT PROBLEM

The main objective of the transport mission is to transport some goods from one point to another in unknown hostile terrain with the help of a robot team. The configuration of the robot team is as follows: a convoy of larger robots carrying the actual goods in need of transport; and a set of smaller robots (scouts) that explore the direct surroundings of the convoy for safety and protection. In the terrain several intruders are present wanting to intercept the payload. The smaller robots will alert the convoy when intruders are detected. At the beginning of the mission a minimal cost plan is calculated for convoy to go from start to goal with the current map information available beforehand. During the mission, as the robots acquire additional information via their sensors, the plan is revised to account for the new information, and thus reduce the total cost of the traverse.

A. Environment Representation.

Figure 1 shows a typical environment for the transport mission. It represents a hilly desert terrain with some streets, buildings and trees.

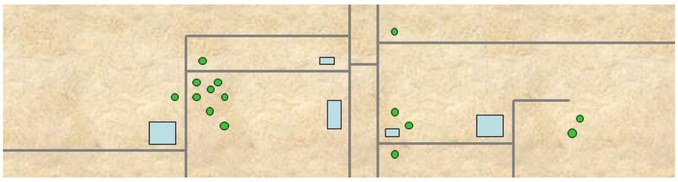

Fig. 1: Typical environment for the transport mission. The map shows a hilly desert terrain with some streets, trees and buildings.

Approximate cellular decomposition is used to model the environment. A map with two 'layers' will be used: one layer representing the entities present in the map, and one that will represent the degree of vicinity to an intruder. Every pixel in each layer is reduced to a grayscale value ranging from 0 to 255 (8 bits).

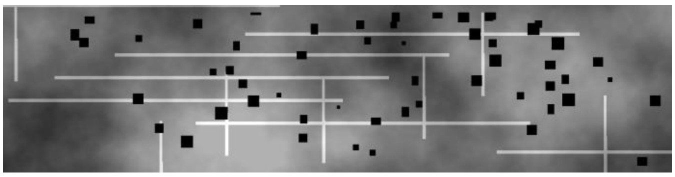

Fig. 2: 8-bit representation of typical environment. Randomly generated.

1) Map

Value '0' is reserved for obstacles; values ranging from 128 to 255 (128 values) are used to describe the height of the street terrain and values ranging from 1 to 127 (127 values) are used to describe the height of off-street terrain. The *traversibility* of the terrain will be in one of 3 conditions: obstacle space (not traversable), off street terrain and street terrain. An example of a map is shown in figure 2.

2) Distance Transform of Intruder Locations

A distance transform maps a binary image into a grey-level image. The binary image here is the location of intruders in an otherwise empty map. The distance transform map changes with the movement of the intruders. The pixel values of the grey level image represent a distance metric from the location of the intruders. This distance transform can be based on different metrics: Euclidean distance, Manhattan distance (city block), chessboard... Here the Euclidean distance is used. Figure 3 shows an example of the distance transform for a snapshot of known intruder position in the terrain. Intruders are show as red dots. The darker regions are more dangerous as they are in closer proximity to the intruders. While planning the path for the convoy the paths preferred will traverse the lighter regions representing the safest places in the map.

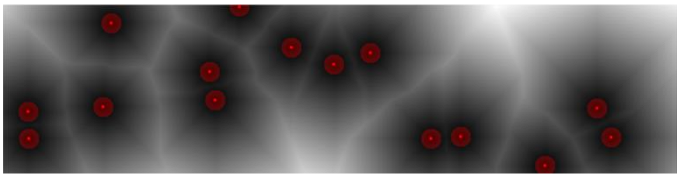

Fig. 3: Distance transform of intruder positions.

B. Intruders

The intruders have as goal to intercept the payload of the convoy. They are all considered identical; with a field of view (FOV) of 360°, a detection range of r_I rand a maximum speed of v_I. They exhibit somewhat intelligent behavior, but they act individually without coordination. Each intruders can perceive everything within its detection range r_I as long as it is not occluded by an obstacle. In figure 4 the intruders and their sensor ranges are depicted in red.

C. Scouts

The objective of the smaller scout robots is to protect and inform the convoy transporting the payload. All small robots are considered identical and have a FOV of 360°, a detection range of r_{scout} and a maximum speed of v_{scout}. They exhibit intelligent behavior in that they act in a coordinated way. Each robot can perceive everything within its detection range r_{scout} as long as it is not occluded by an obstacle. They detect intruders within a range $r_{scoutIDetect} > r_{scout}$ as long as they are not occluded by an obstacle. In figure 4 these robots and their sensor ranges are depicted in green.

D. Convoy

The goal of the convoy, composed of large robots, is to get to the goal position to deliver the cargo as fast and safe as possible. The convoy has a FOV of 360°, a detection range of r_{convoy} and a maximum speed of v_{convoy}. Information gained from the scouts and the convoy itself during the mission is used in deciding what route to take to the goal position. Factors influencing path planning are: proximity of danger, traversibility of the terrain, gradient of the terrain and total distance traveled. As new information comes in, the current planned path will be continuously updated to comply with the mission priorities. In figure 4 the convoy and its sensor range are depicted in blue. The convoy communicates its position with all the scouts.

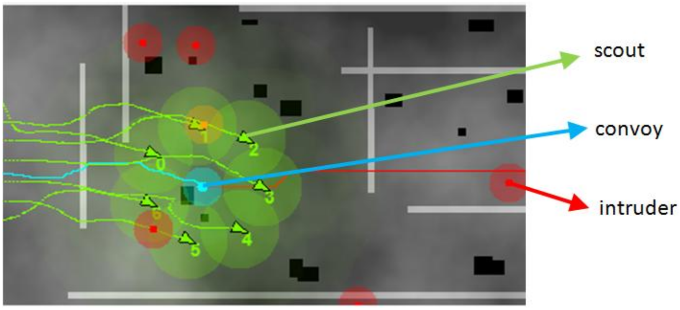

Fig. 4: Screenshot of the simulation: convoy, intruders and scout and their respective sensor ranges. The scout robots try to maximize their information gain by forming a circular formation around the convoy.

IV. Path Planning

A. Local Planning Intruders

The intruders have as goal the interception of the cargo of the convoy. Their default behavior will be to wander around looking for a target to engage with. When a robot enters within its range r_I, the intruder will start pursuit of this robot. When he loses track of this robot he will return to its default wander state. When multiple robots are in the detection range he will follow the one he first detects. In figure 4, robot 1 is being tracked by an intruder.

B. Local Planning Scouts

The task of the scout robots is to protect and inform the convoy during transport. They achieve this by staying close to the convoy and gathering information about its surroundings. This information is used to update the current map information and known intruder locations. The global path for the convoy can thus be revised to steer the convoy to a safer area.

In order to accomplish this task the scouts' movements must be coordinated in such a way as to comply with several constraints simultaneously. Redundancy in information gain must be minimized, i.e. the overlap in the scouts' sensor ranges. This is obtained if they form a circular formation around the convoy each equally spaced from one another, as shown in figure 4. To reduce the chance of an intruder reaching the convoy and minimizing information redundancy, the radius of this circular formation is set to d_{safety}. The safety distance d_{safety} is defined here as the sum of the sensor ranges of the convoy r_{convoy} and scout robots r_{scout}, see figure 5.

1) Behaviours

The scout robots will switch between three behaviours: avoid obstacles (AO), circular formation (CF) and move to convoy (MC). If the distance from a scout to the convoy exceeds a specified threshold distance it is moves back towards the convoy and the MC behaviour is active. Otherwise the CF behaviour is active, which moves the robots into a circular formation. If during any of these two behaviours an obstacle is detected, the AO behaviour activates as long as needed to circumvent the obstacle.

The scout robots communicate their position with every other agent in the robot team; however their observations regarding the environment and intruders are sent only to the convoy.

The movements of the scout robots are controlled through a fuzzy logic controller to achieve smooth transitions between speeds. The outputs of the fuzzy logic controller are speed $\boldsymbol{v}$ and angular speed $\boldsymbol{\omega}$. The goal is to have the scout robots equally spaced in a circle formation around the convoy, moving along with the convoy while maintaining this formation. The scout robots strive to be at a certain distance and orientation of the central convoy at all times.

When the CF behavior is active, a scout robot (R_c) will adjust its speed and angular speed to move towards its goal position (R_c'). This goal position is calculated by looking at its previous (R_p) and next neighbor (R_n) as shown in figure 5. The new goal position is at a distance d_{safety} from the convoy and lies on the angle bisector of the angle α between R_p and R_n.

When the MC behaviour is active the robot will move to the convoy until it is close enough and then switches to the SC behaviour. Consequently only robots close to the convoy spread themselves equally around the convoy. If some robots are blocked or broken along the way a balanced protection is maintained with the remaining robots.

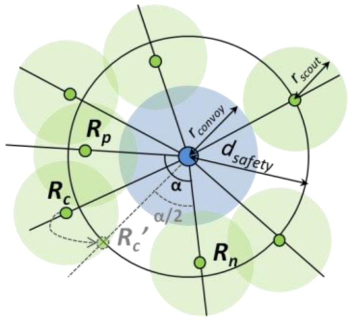

Fig. 5: Every scout robot will adjust its speed v and angular speed ω to move towards its next goal position. This goal position is calculated by looking at the previous and next neighbor R_p and R_n of the current robot R_c. The new goal position R_c' is at a distance d_{safety} from the convoy and lies on the angle bisector of the angle α between R_p and R_n.

2) Fuzzy Logic Controller

A schema of a fuzzy inference system is shown in figure 6. It is usually composed of 5 blocks: a *fuzzification* block which transform the crisp inputs into degrees of membership of linguistic values, a *defuzzification* block which transforms fuzzy values resulting from the inference into a crisp output, a rule-base containing a number of fuzzy *IF-THEN* rules, a *database* which defines the membership functions of the fuzzy sets used in the fuzzy rules and a *decision-making unit* which performs the inference operations on the rules.

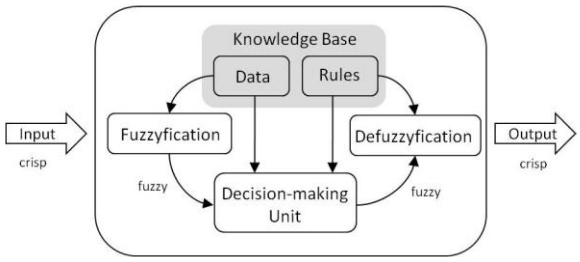

Fig. 6: Fuzzy inference system.

a) Membership Functions

For the control problem at hand a fuzzy logic controller with two inputs and two outputs is employed. The input values are fuzzified by determining the degree of membership in a set of normalized membership function ranging from: NB (negative big), NM (negative medium), NS (negative small), ZE (zero), PS (positive small), PM (positive medium) to PB (positive big), see figure 7. The output membership functions are constants as shown in figure 8.

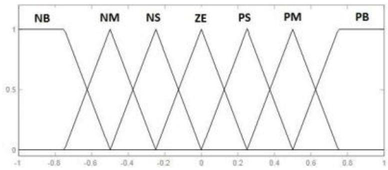

Fig. 7: Membership functions input.

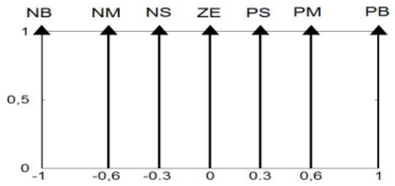

Fig. 8: Membership functions output.

a) Fuzzy Inference

There exist several fuzzy inference methods. The most commonly used is Mandani's [9] inference method. Although in this case a Takagi-Sugeno [10] method of fuzzy inference is used. It is similar to the Mamdani method in many respects. In fact the first two parts of the fuzzy inference process, fuzzifying the inputs and applying the fuzzy operator, are exactly the same. The main difference between Mamdani-type of fuzzy inference and Takagi-Sugeno-type is that the output membership functions are only linear or constant for Sugeno-type fuzzy inference. This will reduce computation time. The fuzzy rules and the corresponding control surface for scout control are given in table 1 and figure 9 respectively.

Table 1: Fuzzy Rules

in1 \ in2	NB	NM	NS	ZE	PS	PM	PB
NB	PB	PB	PB	PB	PM	PS	ZE
NM	PB	PB	PB	PM	PS	ZE	NS
NS	PB	PB	PM	PS	ZE	NS	NM
ZE	PB	PM	PS	ZE	NS	NM	NB
PS	PM	PS	ZE	NS	NM	NB	NB
PM	PS	ZE	NS	NM	NB	NB	NB
PB	ZE	NS	NM	NB	NB	NB	NB

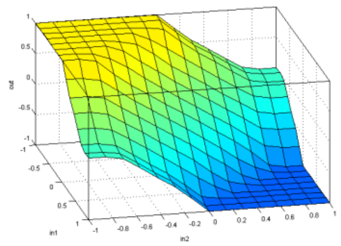

Fig. 9: Control surface of fuzzy rule base in table 1.

3)Spreading around convoy (SC) behavior

The inputs to the fuzzy controller to determine the speed v are d and the derivative of d. d is the distance between R_c and R_c', as shown in figure 10. The output of the fuzzy controller is the change in speed:

$$\Delta v = f\,(d, \Delta d)$$

The inputs for the fuzzy controller to determine the angular speed ω of R_c are the angle θ and the distance d_ω. θ is the angle between the speed of the convoy and the speed of the scout. d_ω is the distance between lines parallel to the speed vector of the convoy passing through R_c and R_c', as shown in figure 10. The output of the fuzzy controller is the new value for the angular speed ω:

$$\omega = f\,(\theta, d_\omega)$$

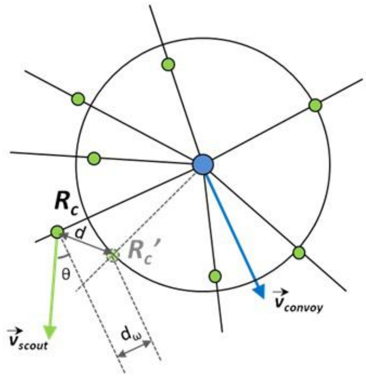

Fig. 10: The inputs to the fuzzy controller to determine the speed v are d and the derivative of d. d is the distance between R_c and R_c'. R_c is the current position of the scout, R_c' is the goal position of the scout. The inputs to the fuzzy controller to determine the angular speed ω are the angle θ and the distance d_ω. θ is the angle between the speed of the convoy and the speed of the scout. d_ω is the distance between lines parallel to the speed vector of the convoy passing through R_c and R_c'.

4)Moving to Convoy Behavior

The inputs for the fuzzy controller to determine the speed v are d and the Δd. d is the distance between R_c and R_c', as shown in figure 11. R_c is the current position of the scout, R_c' is the goal position of the scout. The output of the fuzzy controller is the change in speed Δv:

$$\Delta v = f\,(d, \Delta d)$$

The inputs to the fuzzy controller to determine the angular speed ω are the angle θ and the evolution of θ. θ is the angle between the speed vector of the scout and the line through R_c and the convoy, as shown in figure 11. The output of the fuzzy controller is the change in angular speed:

$$\Delta\omega = f\,(\theta, \Delta\theta)$$

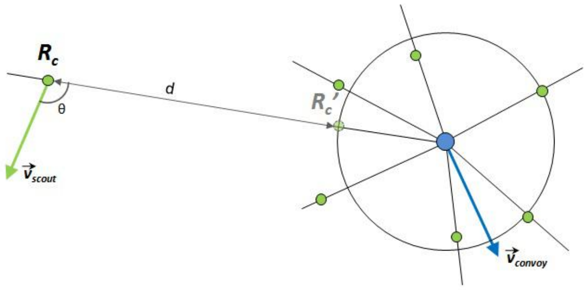

Fig. 11: The inputs to the fuzzy controller to determine the angular speed ω are the angle θ and the evolution of θ. θ is the angle between the speed vector of the scout and the line through R_c and the convoy. R_c is the current position of the scout, R_c*'* is the goal position of the scout. The inputs to the fuzzy controller to determine the speed v are d and the evolution of d. d is the distance between R_c and R_c*'*. R_c*'* lies on the intersection between the line through R_c and the convoy, and the circle around the convoy with radius d_{safety}.

5) Avoiding Obstacle (AO) Behavior

When an obstacle is detected the scout will switch to obstacle avoid mode. By using the local map it has constructed in its memory it performs A * path planning to get around the obstacle.

C.Global Planning Convoy

The transport problem assumes no information about the environment is available. This is a typical situation for outdoor environments, although the location of some streets and larger buildings could be known in advance. With the information available at the start of the mission the convoy plans the lowest cost path to the goal position. During execution of the path the convoy receives constant updates from the scouts about the intruders' locations and the environment. With the intruders moving in the terrain, the environment is constantly changing. In order to handle a dynamic environment and incomplete map information, it is needed that the current plan is updated based on data acquired during the execution of the mission to reduce the total cost of the traverse. As it is crucial that this re-planning must be performed fast; the D* algorithm [10] is used for planning. The D* algorithm plans optimal traverses in real-time by incrementally repairing paths to the robot's state as new information is discovered. D* re-plans only the part of the route which is necessary and reuses information from previous searches to increase speed of operation.

When searching for the global path to the goal different factors are incorporated as listed below. Each will have a specified weight in the calculation. Depending on the primary goal of the mission the weights can be tailored to fit the requirements. They can even be changed during the mission as the focus of the mission changes, triggered by certain events. The obstacles in the scene are extended with the size of the convoy during planning to keep the convoy at a safe distance from the obstacles.

Distance Travelled - The total movement cost is directly proportional to the length of the path. Therefore the convoy will prefer the shortest collision free path.

Gradient Terrain - When intruders are present on or near the streets, the convoy can be forced to deviate from the streets and enter the rougher, off-street terrain. In this situation the cost of the path can be minimized by choosing paths that avoid steep slopes. This cost is comprised in the gradient value of the terrain, which is calculated from neighboring map values that represent the height of the terrain.

Intruder Presence - One of the mission priorities could be for the convoy to stay as far away from the intruders as possible. The darker the regions in the distance transform map of the intruders' locations, the more expensive the cost will be to traverse; as a consequence positions as far removed as possible from the intruders are the preferred regions. This cost is comprised in the distance transform value. Movement and detection of intruders will change the distance transform map during execution.

Traversibility Terrain - In increasing cost to traverse, the possible traversibility states of the terrain are: street, off-street and obstacle terrain. The state, and thus the cost, can be deferred from the map value. As a consequence the convoy will prefer to traverse street terrain, as this is the easiest traversable underground.

The total cost for a path from start to goal is a weighted sum of the different traversal costs that are taken into account while calculating the path for the convoy using the D* algorithm. This total cost must be minimized.

In fig. 12 (a, b, c, d) a snapshot of a mission in execution is shown. In (a) the actual map and intruder position are shown. The part of the path that is already traversed is shown in blue; the part that still needs to be executed is shown in red. In (b) the information about the environment the robot team has gained at this time in execution is shown. (c) Displays the distance transform map of the currently know intruder positions. (d) Shows the distance transform map in convoy memory used to update the path. This information is extracted from (c) that is covered by sensors.

V.SIMULATION RESULTS

The proposed strategy was implemented in C++. Some results are shown that will analyze the effect of path planning weight values on certain mission rating criteria such as path length, number of street deviations and intruder exposure. Intruder exposure is defined as the duration scout or convoy is seen by an intruder. Other factors investigated are the number of scout robots and the size of their sensor range.

Randomly generated maps of size 250 x 1000 similar to the one shown in figure 2 are used. The obstacles and streets are placed randomly. The gradient information is generated with the help of Perlin Noise [11]. Start and goal position are on opposite sides of the terrain.

As can be seen from fig. 12 unknown parts of the map are assumed 'optimistic', i.e. for the distance transform map, what has not yet been perceived is assumed intruder free. For the map of the terrain everything is assumed obstacle free flat street terrain. This results in more exploring during path planning. On the other spectrum ‘pessimistic' assumptions could be made, which limit exploring.

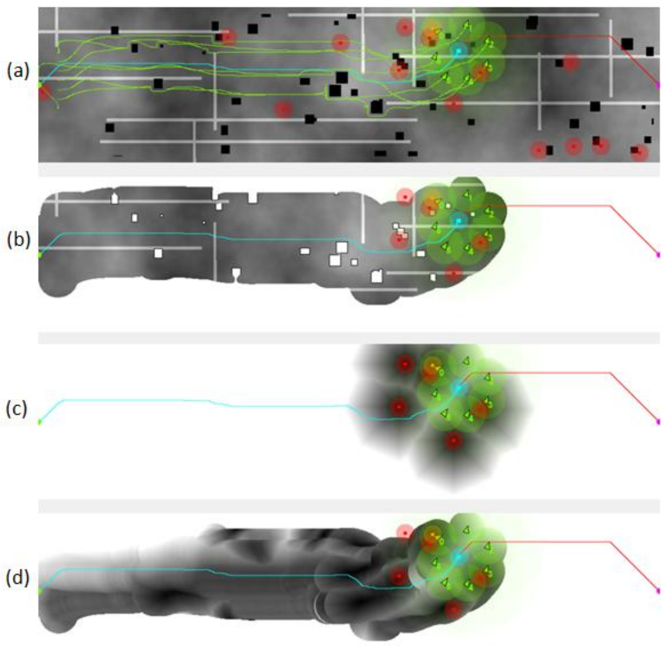

Fig. 12: Snapshot of the mission during execution. In blue the traversed path is shown, in red the current global planned path which changes continuously when new information comes in. (a) The map information, and intruder locations. (b) The information gained by the robot team about the environment at this stage of the mission. (c) The distance transform map of the intruders that are currently in view. (d) Distance transform map in convoy memory used to update the path.

A.Influence Intruder Avoidance

The effect of the weight of the distance transform value on several mission criteria is investigated, and this for a variety of robot team sizes. For each combination of team size and distance transform weight value, the mission results are registered. This is repeated 100 times, the following show the averaged results.

Other parameter settings are kept constant: $v_{convoy} = 1$, $r_{convoy} = 15$, $r_{scout} = 30$, $r_{scoutIDetect} = 3$. r_{scout}, $v_{scout} = 3$, $r_I = 15$, $v_I = 0.5$, $weight_{distanceTravelled} = 1$, $weight_{gradient} = 0$, $weight_{traversibility} = 0$, $nr_{Intruders} = 15$.

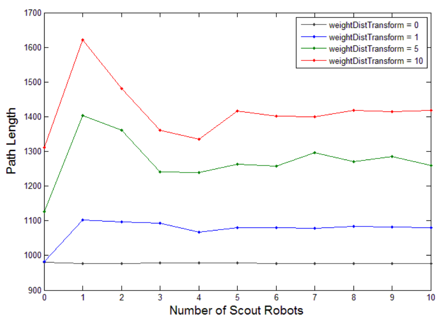

Fig. 13: This graph shows the effect of the distance transform weight value on path length, for different robot team sizes.

The distance transform weight has a major impact on the path length as intruder avoidance will most definitely result in detours increasing the path length. From the graph in figure 13 it can be seen that for increasing distance transform weight the path length increases. With an increase in the number of scout robots, the general trend is that the path length stabilizes. If the distance transform weight is set to zero path length is not affected by an increase in number of scout robots. This is not a general outcome; it comes forth from the nature of the map used in this experiment. Generally path length is reduced with increased view of world.

The peak value for path length for smaller robot teams (1-4 members) is due to optimistic assumptions about the terrain and intruder presence. As there is no complete circular coverage around the convoy for small robot teams, the convoy will continuously try to go the goal via the unperceived parts of the world, leading to longer path lengths.

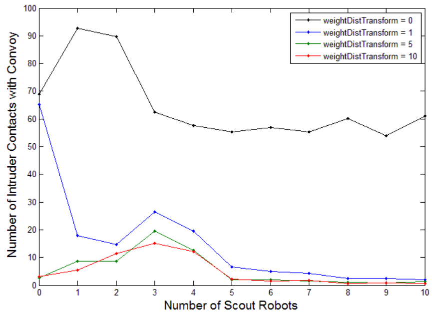

Fig. 14: This graph shows the effect of the distance transform weight value on the number of intruder contacts with the convoy, for different robot team sizes.

In the graph in figure 14 the effect of the distance transform weight on convoy intruder exposure is shown. As to be expected, with an increase in the value of the distance transform weight, the number of contacts the convoy has with the intruders is reduced. For smaller robot teams (1-4 members) an increase is observed in number of intruder contacts, which is the product of two effects. The larger the robot team, the more likely it will attract the attention of the intruders. With a small team the unperceived space between the scout robots is bigger, which allows easier access to the convoy. Increasing the team size further will diminish this effect and result in lesser contacts.

In the graph in figure 15 the effect of the distance transform weight on scout intruder exposure is shown. As to be expected, with an increase in the value of the distance transform weight, the average number of contacts the scout robot has with the intruders is reduced. With an increase in the number of scout robots, the general trend is that the intruder contacts first increase and then decrease. With a small number of scouts the contacts are low as the chance of an encounter is small. As a larger robot team is more ‘visible’ the intruder exposure increases. As the number of intruder contacts plotted is averaged per scout, it will decrease with large robot teams.

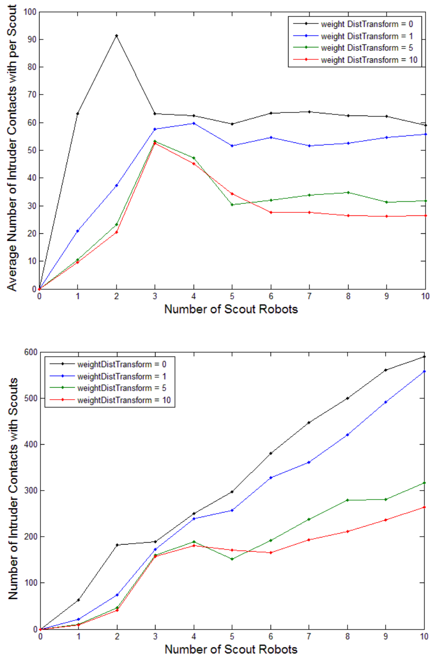

Fig. 15: These graphs show the effect of the distance transform weight value on the number of intruder contacts with scouts, for different robot team sizes. In the top graph the average number of contacts per scout is shown; in the bottom graph the total number of contacts is shown.

B.Influence Scout Sensor Range

The effect of the sensor range value on several mission criteria is investigated, and this for a variety of robot team sizes. For each combination of team size and sensor range value, the mission results are registered. This is repeated 100 times, the following show the averaged results.

Other parameter settings are kept constant: $v_{convoy} = 1$, $r_{convoy} = 15$, $r_{scoutIDetect} = 3$. r_{scout}, $v_{scout} = 3$, $r_I = 15$, $v_I = 0.5$, $weight_{distanceTravelled} = 1$, $weight_{gradient} = 0$, $weight_{traversibility} = 0$, $weight_{distanceTransform} = 1$, $nr_{\ Intruders} = 15$.

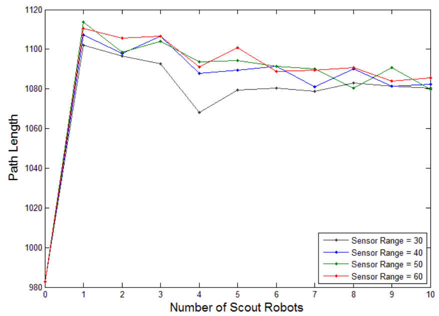

Fig. 16: This graph shows the effect of scout sensor range on path length, for different robot team sizes.

The graph on figure 16 shows the effect of the sensor range on path length for different team sizes. The path length is more or less the same for different values of sensor range, with a slight increase in path length for larger values of sensor range. Although this increase is a consequence of the intruder avoidance behavior: if the sensor range increases intruders are detected faster, which leads to longer path lengths, depending on the strength of the intruder avoidance behavior. Without the intruder avoidance behavior activated, with distance travelled being the sole criteria used for global path planning, an increase in sensor range increases the knowledge about the environment resulting in shorter paths. For maps with larger obstacles, the difference will be more substantial. The jump in path length by adding one member to the team is also due to intruder avoidance.

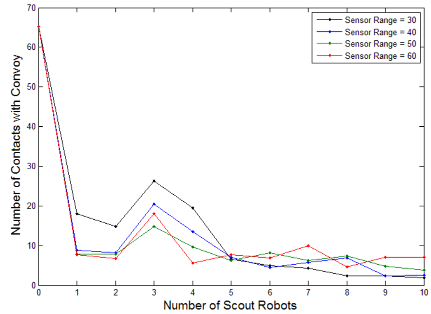

Fig. 17: This graph shows the effect of scout sensor range on the number of intruder contacts with the convoy, for different robot team sizes.

The graph on figure 17 shows the effect of the sensor range on intruder exposure of the convoy for different team sizes. For smaller robot teams a larger sensor range results in less intruder contacts. For large robot teams the number of contacts is rather the similar. With no scout robot team, intruder exposure is significantly larger. It is important to note that with larger sensor range, the distance of a scout to the convoy is larger, as well as the inter-distances between the scout robots, due to increased d_{safety}, which facilitates intruders to reach the convoy. When treating d_{safety} as a parameter, a trade-off can be made between redundancy in information and convoy safety.

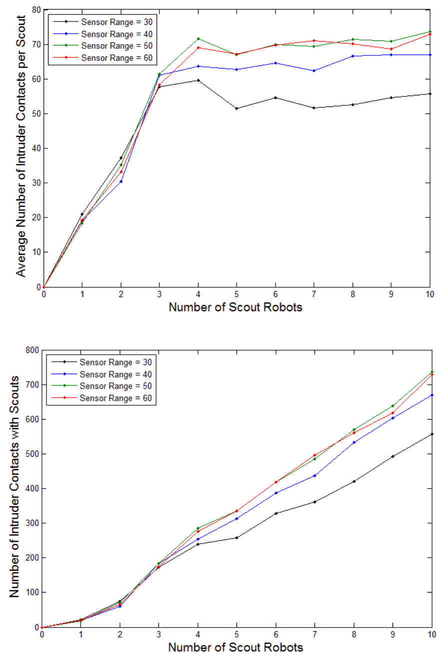

Fig. 18: These graphs show the effect of scout sensor range on the number of intruder contacts with scouts, for different robot team sizes. In the top graph the average number of contacts per scout is shown; in the bottom graph the total number of contacts is shown.

In the graph in figure 18 the effect of the sensor range on scout intruder exposure is shown for different team sizes. It shows that with increased sensor range more intruder contacts occur. With increasing sensor range, the robot team is more geometrically spread, due to d_{safety} as is noted in the previous paragraph. Thus chance on intruder encounter with the scout team increases. The scouts themselves do not perform intruder avoidance, only the convoy does.

C.Influence Street Preference

The effect of the traversibility weight value on several mission criteria is investigated, and this for a variety of robot team sizes. For each combination of team size and traversibility weight value, the mission results are registered. This is repeated 100 times, the following show the averaged results.

Other parameter settings are kept constant: $v_{convoy} = 1$, $r_{convoy} = 15$, $r_{scout} = 30$, $r_{scoutIDetect} = 3$. r_{scout}, $v_{scout} = 3$, $r_I = 15$, $v_I = 0.5$, $weight_{distanceTravelled} = 1$, $weight_{gradient} = 0$, $weight_{distanceTransForm} = 1$, $nr_{\ Intruders} = 15$.

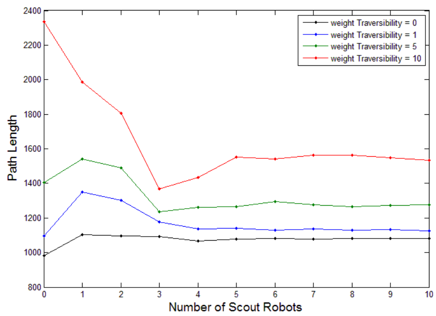

Fig. 19: This graph shows the effect of the traversibility weight value on path length, for different robot team sizes.

The traversibility weight has a considerable impact on the path length, as restricting convoy movement to streets leads to longer path lengths, as shown in figure 19.

With an increase in the number of scout robots, the general trend is that the path length stabilizes. For small robot teams (1-4 members), due to incomplete circular coverage around the convoy and optimistic assumptions about terrain and intruder presence, longer path lengths are registered.

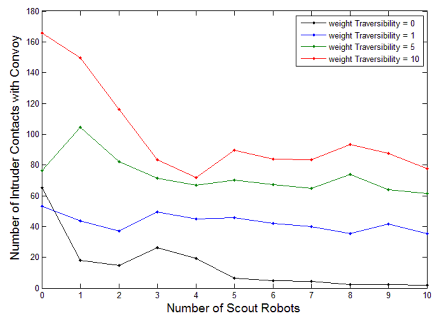

Fig. 20: This graph shows the effect of the traversibility weight value on the number of intruder contacts with the convoy, for different robot team sizes.

In the graph in figure 20 the effect of the traversibility weight on the intruder exposure of the convoy is shown. As increasing the street preference will suppress intruder avoidance, the number of intruder contacts increases with increasing traversibility weight value. For larger robot teams attention gets distracted from the convoy.

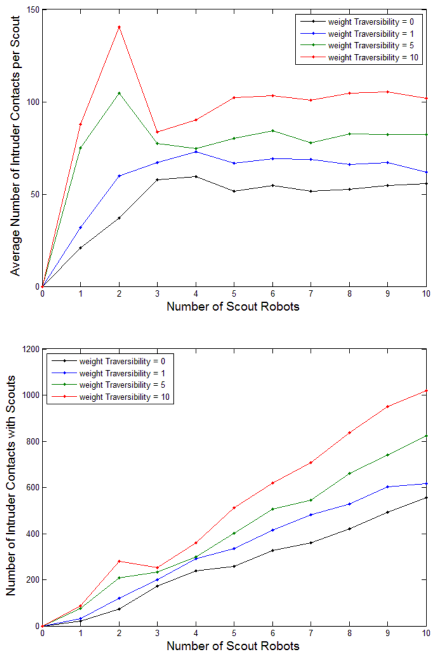

Fig. 21: These graphs show the effect of the traversibility weight value on the number of intruder contacts with scouts, for different robot team sizes. In the top graph the average number of contacts per scout is shown; in the bottom graph the total number of contacts is shown.

Figure 21 shows that with increasing street preference scout robots get more exposed to intruders, for all team sizes. Intruder avoidance is suppressed and the convoy stays on the street.

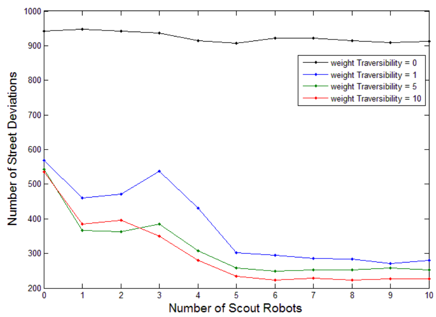

Fig. 22: This graph shows the effect of traversibility weight value on the number of street deviations, for different robot team sizes.

From figure 22 it is deduced that the convoy sticks to street terrain more and more for higher traversibility weight values. For larger robot teams street deviations are even limited further as street terrain is detected more easily.

VI. Conclusion

A strategy was developed to use a cooperating robot team in a transport task. Elements from both centralized and distributed robot control approaches were combined to form a hybrid control structure for the robot team. This results in a more optimal way to complete the task without getting lost in computational complexity. Depending on the specified mission constraints parameters can be tuned to achieve desired behavior.

Acknowledgements

This research was funded by the European Defense Agency (EDA) as part of the Networked Multi-Robot Systems project (NM-RS).

References

[1] Dudek, G., M. & Milios, E. *A Taxonomy of Multirobot System*. In T. Balch & L. Parker, eds, '*Robot Teams: From Diversity to Polymorphism*', A.K. Peters, Natick, Massachusetts, pp. 3-22, 2002.

[2] B.P Gerkey, M.J. Mataric, *A formal analysis and taxonomy of task allocation in multi-robot systems*. In *The Intl. journal of Robotics Research* 23(9):939-954, September2004.

[3] M.B. Dias, R.M. Zlot, N Kalra, and A. T. Stentz, *Market-based multirobot coordination: A survey and analysis.* In *Technical Report CMU-RITR-05-13*, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, April 2005.

[4] Consolini, L., Morbidi, F., Prattichizzo, D., Tosques, M.; *A Geometric Characterization of Leader-Follower Formation Control*, In IEEE International Conference on Robotics and Automation, pp. 2397 - 2402 April 2007

[5] Y.Q. Chen, Z.Wang, *Formation Control: A Review and A New Consideration*, In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3181-3186, Aug. 2005

[6] Coleman, C.P.; Godbole, D., *A Comparison of robustness : fuzzy logic, PID, and sliding model control*, In *Proceedings of the Third IEEE Conference on Systems*, 1994. IEEE World Congress on Computational Intelligence.

[7] Kaynak, O. Erbatur, K. Ertugnrl, M., The *Fusion of Computationally Intelligent Methodologies and Sliding-Mode Control - A Survey*, In *IEEE Transaction on Industrial Electronics*, pp. 4-17, Feb 2001.

[8] E. H. Mamdani, *Application of fuzzy algorithms for control of simple dynamic plant*, In Proc. IEEE, vol. 121, no. 12, pp. 1585-1588, 1976.

[9] T. Takagi, M. Sugeno, *Fuzzy Identification of systems and its applications to modeling and control*, In *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 15, No. 1. (February 1985), pp. 116-132, Feb. 1985.

[10] A. Stentz, *Optimal and efficient path planning for partially-known environments.* In *Proceedings of the IEEE International Conference on Robotics and Automation*, May 1994.

[11] K. Perlin, *An Image Synthesizer*. In *Computer Graphics* (*Proceedings of ACM SIGGRAPH 85*), 24.3.