Reducing Spatial Interference in Robot Teams by Local-Investment Aggression

Mauricio Zuluaga Richard Vaughan
School of Computing Science, Simon Fraser University, Burnaby, BC, Canada
mzuluaga,vaughan@sfu.ca

***Abstract*— This paper extends and improves upon our previous work on the use of stereotypical aggressive display behavior to reduce interference in robot teams, and thus improve their overall efficiency.**

We examine a team of robots with no centralized control performing a transportation task in which robots frequently interfere with each other. The robots must work in the same space, so territorial methods are not appropriate.

In our method, when robots come into competition for floor space, each selects an *aggression level* and the competition is resolved in favor of the more aggressive robot. Our recent work showed that choosing aggression proportional to *task investment* can produce better overall system performance compared to aggression chosen at random. This paper describes a new technique, *local investment*, for computing an aggression level that performs better than any previous method and relies only on local sensor data. The method is evaluated in a simulation study, then shown to be effective in a real-world robot implementation.

***Index Terms*— robot team, interference, aggression, symmetry-breaking, animal behavior**

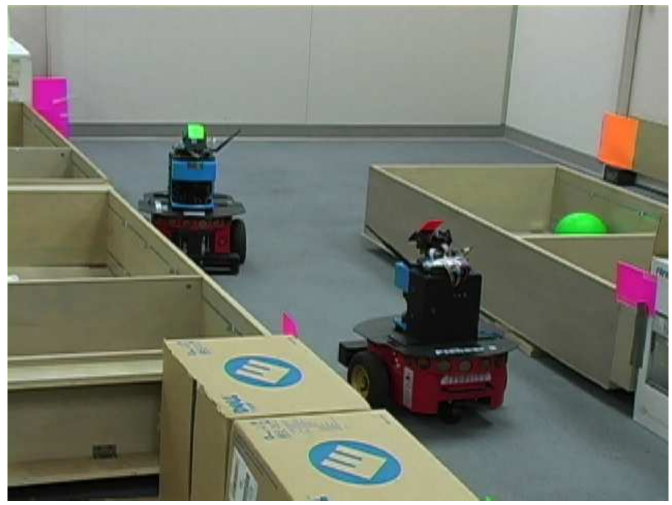

Fig. 1. Spatial interference: two robots block each others' way in a narrow corridor.

I. Introduction

This paper extends and improves upon our previous work on the use of stereotypical aggressive display behavior to reduce interference in robot teams, and thus improve their overall efficiency. Spatial interference is a frequently encountered problem in multi–robot systems, especially those without centralized control, which can seriously degrade performance.

Interference in general can be characterized as competition for resources, for example, access to a charging station or use of a shared tool or sensor. Most commonly, robots simply get in each other's way during normal navigation about the environment. An acute version of this problem is getting two Pioneer–sized (0.5m) robots through a standard (0.8m) doorway from opposite directions; some symmetry–breaking mechanism is required to decide who goes first. This is a real–world problem for robot applications such as mail delivery, factory and warehouse AGVs, and assisted–operator wheelchairs. Another common scenario is shown in Figure 1, where two robots are driving in opposite directions down a narrow corridor, blocking each other's progress.

One possible solution seen in nature is to have the agents fight to determine a winner of the conflict. However, actual robot combat is potentially costly for the individual agents involved and to the overall system as robots may become damaged. The same costs apply in nature and many species have developed aggressive display competitions as abstractions of physical combat to solve interference problems [2], [7], [8].

In a simulated agent–based competition for food resources, [11] has shown that using a display of aggression results in better overall system performance. Their agents were able to perceive the aggression levels of opponent agents and use this information to calculate the cost of fighting for a resource.

Previously, Vaughan et al. in [13] have demonstrated use of a stylized aggressive display, in teams of robots performing a transportation task, to resolve space conflicts in doorways and narrow corridors in a simulated world similar to that shown in Figure 2. The robots must work in the same space, so territorial methods [1], [5] are not appropriate. A key advantage of this system was that communication of 'aggression level' between agents was performed using only the existing navigation sensors and actuators, i.e. there were no special–purpose sensors, no wireless communication and no need for unique identifiers for each robot. Thus the method is perfectly scalable and can be used in heterogeneous systems and even in human–robot interaction; humans can easily understand and manipulate the behavior of the aggressive robots.

The symmetry-breaking provided by the aggressive com-

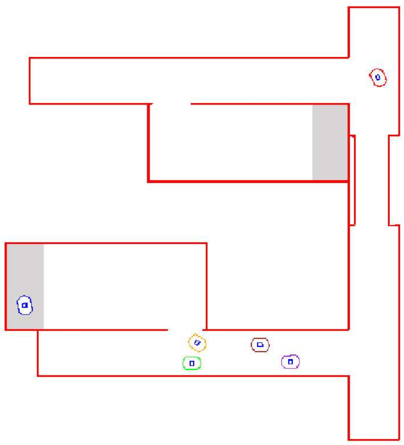

Fig. 2. The Stage world containing six robots used in Experiment 1 and in [4]. It closely resembles the environment used in [13].

petition was shown to produce better overall system performance, in terms of the number of transportation trips completed, compared to an otherwise identical system that lacked the aggression mechanism. Changing the behavior of the robots in this way does not eliminate interference. In a typical one–on–one competition, the 'winning' robot certainly interferes with the immediate progress of the losing robot. Yet as the overall system performance is increased, we may say that the overall negative interference is reduced.

Several strategies for determining a robot's aggression level during a 'fight' were evaluated: random aggression; a linear dominance hierarchy; and a 'personal–space' method, where aggression was determined by the amount of free space visible to the robot. All methods were shown to have statistically similar performance. Neither the dominance hierarchy nor the personal space method offered any improvement over a random outcome.

In [4] we introduced a principled approach to selecting an aggression level, based on a robot's investment in a task. The concept of 'investment' of work done towards achieving a goal is fundamental in models of autonomy in animals [9]. Simulation experiments with teams of six robots in an office–type environment showed that, under certain conditions, this method was able to significantly improve system performance compared to a random competition and a non–competitive control experiment.

This paper describes a new approach to selecting the aggression level based on the concept of 'local investment'. In this method, aggression level is proportional to the effort that a robot has put recently into crossing areas where there is a high probability of interference. This method was designed to overcome a limitation of the previous 'global' investment method. The rest of this paper describes the method and demonstrates that it performs better than the previous known best method, both in simulation experiments and in a smaller-scale real-robot implementation.

II. Rational Aggression

To improve performance compared to a random outcome, the outcome of an aggressive interaction must reflect some relevant state of the world. As argued in [13], a hierarchy of robots with fixed aggression levels can not encode any information relevant to the outcome of a particular competition; when two robots meet at a doorway, their status in the hierarchy does not matter, so long as one of them gives way. Adding memory of past robot/robot interactions does not help; similar arguments apply to dynamic hierarchies.

In general, to control some parameter of a system, we must measure it or estimate it from its correlates. To maximize the amount of work done by our robot system, we need an estimate of how much work a robot is doing as an input into our control system. This principle leads to the following economic approach to this problem.

Consider a system of two robots, Black and White, working in a narrow corridor as shown in Figure 3. They have the same task; transporting widgets from A to B at either end of the corridor. Assume that it is not practical for robots to transfer widgets between themselves. Black starts at A, White at B. At some moment, shown in the top row of the figure, Black and White block each other's progress. Assume the robots have an internal aggression level, and can perform a stereotypical behavior sequence called a *fight*, in which each robot displays its aggression to the other. If a robot perceives that its rival has a higher level of aggression, it goes into retreat mode and allows the other robot to move forward. By performing a *fight*, Black and White can resolve their conflict; the more aggressive robot will push the other backwards and out of the way. From the point of view of system efficiency, which robot should be more aggressive?

A. Global Investment Aggression

In Figure 3 the arrows beneath the robots indicate how far the robots have travelled towards their goals. This travel inevitably has real cost in terms of time, energy and computation. These are *sunk costs*; they can not be recovered. In the left column, Black wins the fight and pushes White along the corridor until Black reaches its goal (middle row). Then Black switches to goal A, and proceeds down the corridor followed by White. At some point (bottom row), White is now back where it started to fight, after travelling the distance indicated by the arrow. The cost of the fight is the sum of White's sunk cost plus the cost to get back to its start position. The right column shows the outcome if White wins the fight. The steps are the same, but the total cost of Black losing the fight (total length of arrows) is much smaller. In this thought experiment, the robot with the higher sunk costs should be more aggressive as it has more to lose. With this scheme, the system will achieve more trips from A to B in unit time than with randomly chosen aggression. The method is economically rational; it makes decisions based on the expectation of a favourable outcome. We will refer to this as the 'global investment' method.

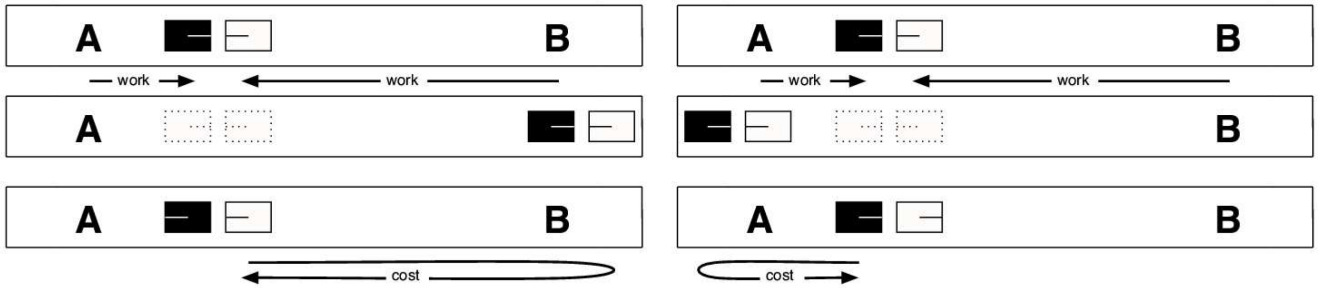

Fig. 3. Motivation for the *global effort* strategy. The robot that has invested the most work in a task should win.

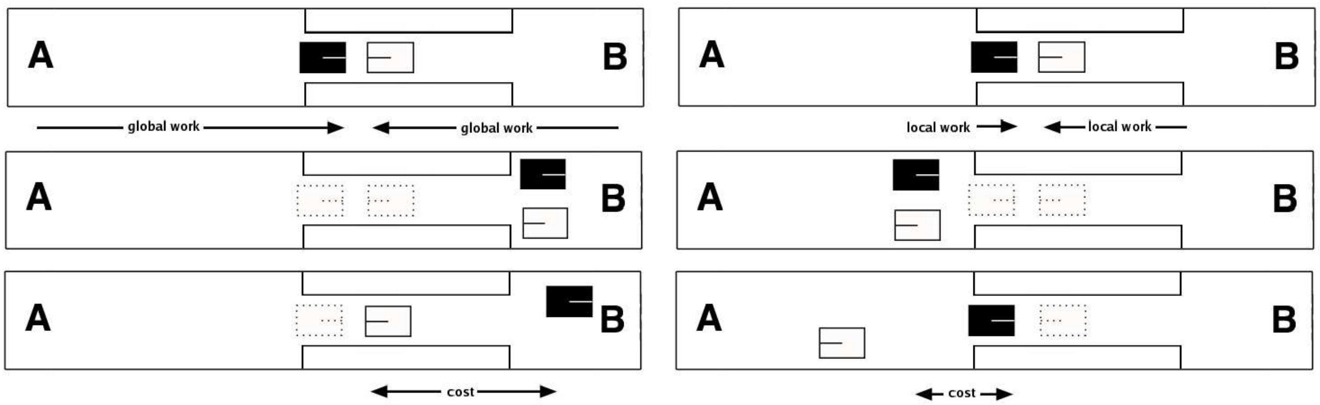

Fig. 4. Motivation for the *local effort* strategy. The robot that has invested the most work passing the narrow part of the corridor should win.

The global investment method can be implemented very simply by adding a minimal memory to the robot: a counter. The counter is incremented each control loop cycle. On reaching a goal the counter is reset to zero. The value in the counter reflects the amount of time the robot has spent on reaching its current goal. The aggression level is set proportional to this value.

B. Local Investment Aggression

The local investment method is also economically rational, and similar to the global investment approach but instead of looking at the total effort put into finishing a task, we only look at the effort put into passing an area in which interference is likely.

Now consider the system of robots presented in Figure 4. The task is the same as in the global approach explained previously. The black robot transports resources from A to B and the white robot from B to A. In about two thirds of the path the corridor is wide enough to allow two robots to pass without interfering with each other. There is however a narrow part of the corridor (about one third of the total length) wide enough for one robot to pass.

Figure 4 presents the two possible outcomes of a conflict happening in the left part of the narrow corridor. Either the white robot retreats and allows the black robot to pass or the black robot retreats and the white robot moves forward. Once both robots are in the wide part of the corridor they can resume navigation without any interference.

It is seen in the figure that the sunk costs are higher for the white robot loosing the fight. Therefore we would prefer a fight where the white robot wins the right of way. In the situation presented in this figure, a global investment approach would choose in favour of the black robot, because it has put more work into getting to its current location compared to the white robot (left column of the figure). On the other hand the local investment approach would allow the white robot to win, as it has spent more time inside the narrow part of the corridor (right column of the figure). Therefore, in this situation the local investment approach handles the conflict in an optimal way.

Figure 5 shows two examples of real-world fights, implemented on Pioneer 3-DX robots. The top row shows the sequence (left-to-right) of a fight with a favourable outcome, in which the interference is minimized because the robots back up as little as possible. The bottom row shows an unfavourable fight, in which one of the robots has to back up a long way before resuming its normal path.

III. Experiment 1: Local Investment *vs* Global Investment and Random Aggression in a Simulated Environment

This section describes the experiments carried out to compare the performance of the three methods when used in a team of robots. We follow the same experimental design and

Fig. 5. Example of two different fights. The upper row shows a fight sequence in which the cost is minimized. The lower row shows a fight sequence in which the cost is very high (one robot was close passing through an area of interference and yet had to retreat.)

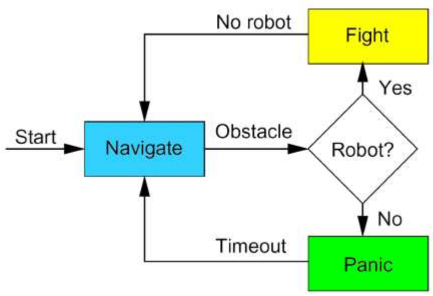

Fig. 6. Control architecture.

use the same robot controller in [4]. We compare the local investment approach together with the best approaches found in [4]: global investment and random.

A. Task

Robots have the mission of transporting resources back and forth between two goal locations (shaded areas in the rooms, Figure 2). Our starting condition is the same used in [4], that is, when a trial is executed all robots go to the same starting position. Note that the world in Figure 2 closely resembles the world used in [13] therefore our results can be compared with those described in that paper.

B. Control Architecture

Each robot runs the same control program, shown schematically in Figure 6. Each mode is described below.

1) Navigate: Navigate is the default mode for the robot and executes until an obstacle in the path is found. The implementation of navigate is an adaptation of the controller in the previous papers. Instead of using a crumb trail to navigate, our controller utilizes a global map which, for any point in the world, provides the correct general direction for the robot. The direction vectors change depending on which goal area the robot is currently seeking.

Robots perform left wall–following and obstacle–avoidance using the robust 'sliding box' algorithm described in [13] and [4]. We omit the details here for space reasons.

Other robots in the system are not treated as obstacles, i.e. robots should not try to navigate around each other. To prevent this, robots are erased from the laser scan used for navigation. The gaps in laser data are filled in by interpolating from the rest of the scan.

The wide corridors in the simulated world have enough space for two pioneer robots to pass comfortably. When turning corners, however, robots that swing too wide, or not wide enough, can interfere with each other.

2) Panic: The purpose of *panic* is to move robots away from situations in which they are stuck, either because of obstacles or other robots interfering. Panic initially causes the robot to simply sleep for a random number of cycles in case the problem goes away. On awakening, if the obstacles are still present, the robot will first attempt to rotate for a random length of time and if unable to rotate, the robot will try to move backwards slightly. Moving backwards provides space for robots which are too close to walls or obstacles to turn without hitting any objects. Robots stay in the panic cycle and switch between sleeping, rotating, and moving forward until they are able to navigate forward successfully for a short period of time.

3) Fight: The *fight* procedure is triggered when a robot emergency–stops and detects another robot in front. Usually this is because the other robot is blocking the way. When entering the *fight* mode, the robot calculates its *fear threshold*: the minimum distance it will tolerate to another robot. The robot starts fighting by moving backwards; it continuously backs away as long as it detects another robot within its fear threshold. If the rival is outside its fear threshold, the robot switches back to the *navigate* mode. If the robot is too close to an obstacle while it is moving backwards, the emergency stop mechanism is invoked. In this case the robot stops and switches to the *panic* procedure. During a fight the robot with the smaller fear threshold will be the first to start driving forward again - this robot is the *winner*. Consequently, it will push its rival (the *loser*) backwards until there is enough room to pass. Once the winner moves outside the loser's fear threshold, for example by passing, the loser starts navigating again and the fight is over.

C. Aggression Function

The fear threshold is determined by a robot's *aggression* α, where $0 < \alpha < \alpha_{max}$ and $\alpha \in \mathbf{R}$, selected at the start of the *fight* procedure. *Fight* is designed so that the more aggressive robot is likely to be the winner.

The fear threshold is the minimum distance one robot can tolerate to another robot and is inversely proportional to a robot's aggression, plus some offset distance.

$$\phi = K_1 + \frac{K_2}{\alpha} \quad (1)$$

Constants K_1 and K_2 were chosen to give a fear threshold between 450mm and 2450mm in our experiments. A tiebreaker mechanism is employed in order to reliably resolve the fight between two robots with very similar fear thresholds. It adds a small random distance to a robot's fear threshold, ensuring that two robots with the same aggression have different fear threshold values. It breaks the symmetry between two robots by preventing them from switching from *fight* to *navigate* at the same time.

In this paper, we compare three different aggression functions:

1) Random: Aggression α is chosen at random in the range $0 < \alpha < 10$.

2) Global Investment: Aggression value is proportional to the time a robot has spent approaching the goal in each trip. A robot's aggression increases with the time it has spent on the *navigate* behavior within the current trip. Specifically, we calculate the aggression A using the formula:

$$\alpha = min(K_3 \frac{T}{T_{normal}}, \alpha_{max}) \quad (2)$$

where T is the time spent approaching the current goal, T_{normal} is a normalization constant reflecting the expected time to reach the goal, and K scales the aggression to the desired range. α_{max} sets the upper bound of α, so that $0 < \alpha < \alpha_{max}$.

3) Local Investment: Aggression is random if outside an area of interference, and proportional to the time spent inside an area of interference if inside an area of interference. We calculate the aggression using the following formula:

$$\alpha_{in}(t) = m_{in} \cdot t_{in} + C_{in} \quad (3)$$

$$\alpha_{out}(t) = C_{out} - m_{out} \cdot t_{out} \quad (4)$$

$$\alpha_{rnd}(t) = m_{rand} \cdot random(0 \ldots 1) \quad (5)$$

Aggression α is equal to $\alpha_{in}(t)$ if inside an area of interference, equal to $\alpha_{out}(t)$ if we are just comming out of an area of interference and equal to $\alpha_{rnd}(t)$ if $\alpha_{out}(t)$ becomes negative. In our experiments $C_{in} = 0.6$ $C_{out} = 1.0$ $m_{in} = 0.008$ $m_{out} = 0.03$ and $m_{rnd} = 0.03$. The final value of aggression was also thresholded between 0 and 1.

Figure 7 shows a graph of a robot's aggression while moving left to right in the environment. The aggression is initially random because the robot is not in an interference area. Then as soon as the robot starts to move inside the narrow corridor its aggression starts to increase linearly until reaching the maximum ($\alpha = 1.0$) around the end of the narrow part of the corridor. Once the robot is out of the corridor its aggression decreases quickly and when below some threshold it reverts to random aggression. Note that the random aggression is always smaller than the minimum aggression inside the corridor therefore a robot inside the corridor fighting with a robot outside the corridor is more likely to win the fight.

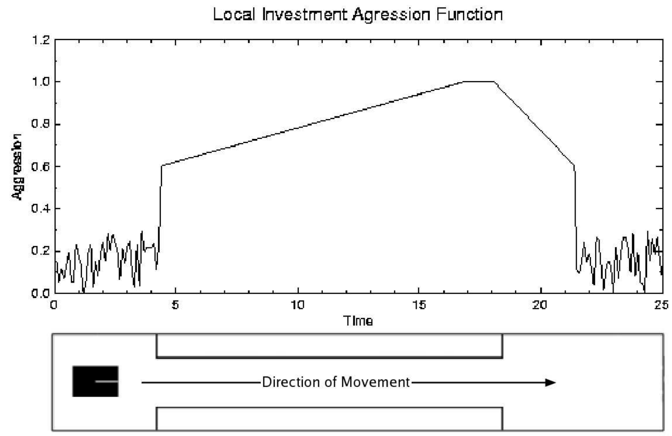

Fig. 7. Local Investment Aggression

D. Procedure

There are 6 robots living in the world shown in Figure 2. The world has rooms and corridors where the robots can move, the doors and some sections of the corridor are narrow and only allow one robot to pass by. We ran 24 trials that lasted 1800 seconds for each aggression function (local investment, global investment and random).

Each time we started a new trial, the location of the robots was reset to the same initial position.

The simulation environment used is the well-known Player/Stage system [6].

As soon as a trial began, all robots began to log information regarding the type of aggression function used, the trial number, the robot number, the number of trips completed, and the total time spent in *navigate*, *fight* and *panic* behaviors.

E. Performance Metric

To measure the success of a trial we count the total number of trips T_{team} performed by all the robots

$$T_{team} = \sum_{i=1}^{n} T_i \quad (6)$$

where n is the number of robots in the team and T_i is the number of trips performed by robot i.

This value is easy to obtain and represents an objective measurement of the performance of the system as a whole. In our resource transportation task, each trip completed by a robot is equivalent to one unit of resource transported. We are not trying to improve the number of trips that a single robot does but rather the number of trips that the team of robots complete. With this metric, we obtain a single score with which to compare the performance of the different aggression functions.

F. Statistics Tests

The primary statistical test used to evaluate the performace of the three different aggression functions was a T–test with significance < 0.05. The T-test shows whether the means for the number of resources transported by each of the aggression schemes is significantly different. A few trials were considered to be *outliers*, that is, their total resources transported values differed greatly from the rest of the data collected. We used 1.5 times the interquartile range (IQR) as fences and those trials which fell outside of the fences were considered outliers and removed from the data set.

G. Results

The results presented in Table I show no improvement in performance using the global investment based aggression scheme over the random scheme. However the local investment strategy shows a significant difference when compared to the random and global investment schemes. This can also be seen in the shape of the distributions presented in Figure 8. While random and global investment have similar distributions, the local investment is shifted to the right.

Note that the standard deviation of the local investment seems large compared to the random and global trials, this is because some trials are marked as outliers for random and global investment, and in this way their standard deviations become smaller. The removal of the outliers did not greatly affect the significance of the T-tests in any case.

The good results obtained by the local investment approach are caused by two properties. The first is that, as intended by the method, we observe that fights occurring in the narrow part of the corridor are won by the robot who has the most to lose. This small advantage accumulated over time and over multiple fights makes a significant difference in the performance of the team. Also, there is an emergent property of the system that is generated by the local investment approach: robots quickly form "chains" or "worms" going in the same direction. Once you have a worm created, the frequency of interference is reduced.

We also performed a second experiment using the local investment method in a second environment described in [4]. In this experiment the world is similar to the one shown in Figure 2 but it has longer areas of interference and bigger rooms. The results we obtained with the local investment approach were statistically similar to those obtained by the global investment approach and better than the ones using the random aggression mechanism. These results can be explained by observing that, in a world with very large areas of interference, the local investment method closely approximates the global investment method.

IV. Experiment 2: real world robots

A second experiment was performed in order to (i) verify that the aggressive display technique is feasible for real robots, and (ii) to demonstrate the effectiveness of the local investment strategy in a real-world implementation.

This experiment was done in two parts: a simulation to test the experimental procedure and obtain benchmark results, followed by the real-world trials. This experiment was smaller-scale than the simulations because only two Pioneer robots were available.

A. Task

The robots must perform laps of an 'O' shaped world, shown in Figure 9, the dimensions of the world are 2.85 by 8.5 meters. The narrow corridors are 1 meter wide and the inner block dimensions are 0.85 by 3.5 meters. These dimensions are the same for the simulated and real world. The robots complete loops of the world, returning to their start position on each loop. The robots go opposite directions and therefore interfere with one another very frequently. Given the shape of the world, the robot going in clockwise direction makes big loops while the robot going in counter clockwise direction makes small ones. It takes 45 seconds for one robot to complete the small loop, while it takes 66 seconds for the other to complete the big loop.

The 'O world' is designed to cause frequent interference. It also increases the penalty of the wrong robot losing a fight, as the areas of interference constitute the majority of the world (long corridors). In practice we found that in 20 minute trials with two robots we obtain an average of 30 fights. These characteristics allow us to show a difference in performance when using each of the different aggression functions.

The starting condition for each of the trials is always the same, as shown in the simulated world in Figure 9 (left).

B. Control Architecture

The details of the controller used on the 'O world' experiments are different from the one used in the earlier simulation experiments. However, the same general control architecture is used (Figure 6). The more simple structure of the 'O World' permitted a simplified controller, which proved to be robust; the robots were completely autonomous for the length of the trials.

1) Navigate: Given the shape of the O world, a left wall follower is all that is needed to navigate through the environment. Robots do not use a map or any localization information.

2) Panic: The panic behavior is also simplified to two sub behaviors: a 'corrective panic' which corrects the heading of a robot when, for example, the robot does a turn and gets to close to the wall. The response is then just to move in the opposite direction and turn a little. If this is not enough to solve the problem then the sonar is used as an array of forces acting on the robots and proportional to the distance. The robot moves in the direction of the resultant force.

3) Fight: This behavior is responsible for deciding how to solve interference problems between robots.

The fighting behavior we used in the first simulation experiments presented at the beginning of the paper did not work well in the real world because it required too much free space behind the robots in order to decide which robot won a conflict. This is because the aggression level of a robot was used to generate a tolerance distance. But two different aggression levels needed to be perceived as two different distances, if we had 10 aggression levels and for example 0.4 meters to differentiate between each one, then we would require at least 4 meters plus some safety distance in order to allow the robots to detect who won a fight safely. Reducing this distance causes the robots to get into oscillations. That is, two robots with similar aggressions start to move back and forth because they both think they won or lost a fight. Another problem of the method is that it wastes energy because the robots have to back up a long distance. The 'O World' was too small for this technique to be used.

To solve the problem we developed a alternative "staring contest" scheme in which the aggression level is converted to a waiting time. The method is simple: when two robots going in opposite directions find the other robot too close they both stop. They then use their aggression level to calculate how much time they are willing to wait before retreating (losing the fight). While waiting they are constantly checking the distance to the other robot. If they perceive that the robot is backing up then they know they won the fight and move forward. This method has two benefits over the previous back-up fight: it uses less energy and it requires less space.

C. Performance Metric

In the previous experiment the work done by the team of robots was directly proportional to the amount of trips they completed. In this case however we have to be more careful given the difference in time or energy expended to complete a big or a small loop. We have chosen the completion of one small loop to be equivalent to 1 unit of work. Using the difference in time (effort) that takes to complete the big and the small loops, we can obtain a conversion factor and calculate the total work done by a team of robots with the Equation shown in 8. This enables us to compare the results of different trials and different aggression functions with one another.

$$work_{cf} = \frac{time_{big}}{time_{small}} = \frac{66}{45} = 1.46 \qquad (7)$$

$$work = Trips_{small} + Trips_{big} \cdot work_{cf} \qquad (8)$$

1) Simulation Results: In the simulation test we ran a total of 20 trials of 20 minutes for the random, global and local investment aggression functions.

The simulation results in Table II show that while the random and global investment approaches perform similarly, the local investment approach is better. Even the number of trips completed by the robots using local investment aggression is greater than with random and global investment.

Having shown that the performance of global investment is indistinguishable from random in this environment, we chose to compare only local investment and random in the real robot trials.

TrialType	Mean Resources	σ	N	Outliers
Random	65.54	7.93	24	2
Global Investment	63.57	9.8	24	3
Local Investment	75.29	14.22	24	0

TABLE I

EXP1 RESULTS: SUMMARY OF PERFORMANCE SCORES FROM THREE DIFFERENT AGGRESSION FUNCTIONS.

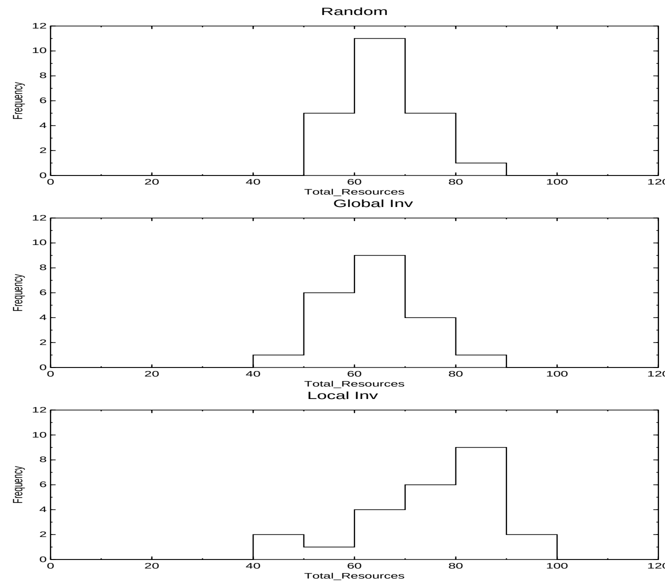

Fig. 8. Exp. 1 results: Histograms showing distribution of performance scores for three different controllers: random aggression, global investment and local investment aggression.

Fig. 9. Simulated (left) and real-world (right) environments. Only one robot at a time can pass across the narrow corridors on the left and the right.

Trial Type	Total Work	Total Trips	$R_{bigloop}$ Trips	$R_{bigloop}$ σ	$R_{smallloop}$ Trips	$R_{smallloop}$ σ
Random	34.4	30	9.6	0.59	20.4	1.14
Global Inv.	34.2	28.8	12.0	0.9	16.7	1.2
Local Inv.	41.7	34.7	15.4	0.5	19.3	1.37

TABLE II

SIMULATION EXPERIMENT IN THE O WORLD RESULTS: SUMMARY OF PERFORMANCE SCORES FROM THREE DIFFERENT AGGRESSION FUNCTIONS.

Trial Type	Total Work	Total Trips	$R_{bigloop}$ Trips	$R_{smallloop}$ Trips
Random 1	29.1	25	9	16
Random 2	31.6	27	10	17
Random 3	29.6	25	10	15
Random Avg	30.1	27.3	9.6	16
Local Investment 1	39.3	32	16	16
Local Investment 2	38.3	31	16	15
Local Investment 3	37.9	31	15	16
Local Investment Avg	38.5	31.3	15.6	15.6

TABLE III

EXPERIMENT IN THE REAL WORLD RESULTS: SUMMARY OF PERFORMANCE SCORES FOR RANDOM AND LOCAL AGGRESSION.

2) Real World Results: In this experiment we ran a total of 3 trials of 20 minutes for the random, and local Investment aggression functions.

Table III presents the results obtained in the real world. Due to the small sample size, we do not attempt to give standard deviations or histograms.

However, we observe that the results are similar to those obtained in simulation, and that aggression based on local investment always produces more trips overall than random aggression.

V. LIMITATIONS AND FUTURE WORK

Despite its success, the local investment scheme has the main limitation of being local and therefore blind to what is going on in the whole world. It does not know anything about the location of other robots outside its sensor range and cannot make decisions based on any global properties. This limitation is shared by the previous methods we have studied, and is discussed further in [4]. This problem notwithstanding, for many real situations the local investment method can help to improve the performance of a team of robots by reducing interference.

We plan to extend this work to allow robots to choose their aggression function adaptively, depending on the properties of the world in which they find themselves. Also, we are attempting to automatically learn novel aggression functions: they will be compared to 'local investment' as the current best performer.

We are also investigating methods in which sub-groups of robots, such as the 'worms' mentioned above, can combine their aggression using communication so that groups will win fights with individual robots. Our expectation is that this may provide a further significant improvement in performance without requiring global information or communication.

As performance improves, it is becoming more difficult to show statistically significant differences between aggression functions. With this in mind, we are developing more sophisticated navigation controllers in order to demonstrate these techniques in more complex worlds with large numbers of robots.

VI. CONCLUSION

In Summary, We have extended previous work on reducing interference in robot teams by devising a novel aggression function, based on the idea of local investment. An implementation was described, and the method was shown to increase the performance of a simulated robot team at a classical transportation task and, in a simplified form, in real life with a team of two robots. Though this method has some limitations, it should be widely applicable as an interference reduction technique in mobile robot teams.

ACKNOWLEDGEMENTS

Thanks to Sarah Brown, Carl Zhang and Jens Wawerla for contributing to the ideas developed in this paper. This work was supported by an NSERC Discovery Grant to Richard Vaughan.

REFERENCES

[1] R. C. Arkin and T. Balch. Cooperative multiagent robotic systems. In D. Kortenkamp, R. P. Bonasso, and R. Murphy, editors, *Artifi cial Intelligence and Mobile Robots*. MIT/AAAI Press, Cambridge, MA, 1998.

[2] Robert Axelrod. *The Evolution of Cooperation*. Basic Books, New York, NY, 1984.

[3] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2:14–23, 1986.

[4] Sarah Brown, Mauricio Zuluaga, Yinan Zhang, and Richard T. Vaughan. Rational aggressive behaviour reduces interference in a mobile robot team. *Submitted to The 12th International Conference on Advanced Robotics, available on-line at http://www.cs.sfu.ca/research/groups/autonomy/doc/brown_icar05_submitted.pdf*, 2005.

[5] M. S. Fontán and M. J Matarić. Territorial multi–robot task division. *IEEE Transactions on Robotics and Automation*, 14(5), 1998.

[6] Brian P. Gerkey, Richard T. Vaughan, Kasper Stoy, Andrew Howard, Gaurav S. Sukhatme, and Maja J. Matarić. Most valuable player: A robot device server for distributed control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1226 – 1231, 2001. (Also appears in Proceedings of the Second International Workshop on Infrastructure for Agents, MAS, and Scalable MAS at Autonomous Agents 2001, Montreal, Canada, May 29, 2001).

[7] J. R. Krebs and R. Dawkins. Animal signals: mind reading and manipulation. In J. R. Krebs and N. B. Davies, editors, *Behavioural Ecology: an evolutionary approach*, pages 380–402. Blackwell, 1984.

[8] John Maynard-Smith. *Evolution and the Theory of Games*. Cambridge University Press, 1982.

[9] D. McFarland and T. Bosser. *Intelligent Behaviour in Animals and Robots*. MIT Press, 1993.

[10] David McFarland. *Animal Behaviour*. Longman Scientific & Technical, 1985.

[11] Matthias Scheutz and Paul Schermerhorn. The more radical, the better: Investigating the utility of aggression in the competition among different agent kinds. In *Proc. of SAB 2004, Los Angeles, CA*. MIT Press, 2004.

[12] Richard T. Vaughan, Brian P. Gerkey, and Andrew Howard. On device abstractions for portable, reusable robot code. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2121–2427, Las Vegas, Nevada, U.S.A, Oct 2003. (Also Technical Report CRES-03-009).

[13] Richard T. Vaughan, Kasper Stoy, Gaurav S. Sukhatme, and Maja J Mataric. Go ahead, make my day: robot conflict resolution by aggressive competition. In *Proc. Int. Conf. Simulation of Adaptive Behaviour, Paris, France*, 2000.