To appear in *Proc. IEEE Int. Conf. Robotics and Automation*
Orlando, Florida, USA, May 15-19, 2006

Ganging up: Team-Based Aggression Expands the Population/Performance Envelope in a Multi-Robot System

Yinan Zhang
School of Computing Science
Simon Fraser University
yinanz@sfu.ca

Richard Vaughan
School of Computing Science
Simon Fraser University
vaughan@sfu.ca

***Abstract*— We examine a team of robots with no centralized control performing a transportation task in which robots frequently interfere with each other, thus impairing overall team's performance. It has previously been shown that stereotyped robot–robot competitions, inspired by aggressive displays in animals, can be used to effectively reduce interference and improve system performance for this task. We describe an extension to the previous best–performing 'aggression function' to dynamic teams of robots. Experimental results show that the new method provides the best performance yet seen. Further, we examine the effects of interference–reduction methods over a range of population sizes, and we compare the results to a previously suggested theoretical model.**

I. Introduction

One problem frequently encountered in multi–robot systems, especially those without centralized control, is interference between agents. 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: a 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.

One possible solution seen in nature is to have the agents fight to determine a winner of the conflict. However, actual robot combat is costly for the individual agents involved and to the overall system as robots may become damaged. The same costs also apply in nature, and many species have developed aggressive display competitions as abstractions of physical combat [1], [7], [6] to solve this problem.

Vaughan, et al. [8] demonstrated 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 1. This approach is best categorized as eco–problem solving, as described by Ferber in [4]. Instead of considering the multiple–agent system as a whole, eco–problem solving simplifies the complexity of the system by decomposing it into small independent entities, called eco–agents. The overall objective of the entire system is satisfied if all of the eco–agents within the system achieve their goals by interacting with each other locally. A key advantage of this aggressive display method is that each agent displays its 'aggression level' using only the sensors and actuators it uses for navigation; no symbolic communication occurs between robots. Therefore, the system is completely decentralized, perfectly scalable, and can be used in heterogeneous systems and even in human–robot interaction: humans can easily understand and manipulate the behaviour of the 'aggressive' robots.

The symmetry–breaking provided by the aggressive competition was shown to produce better overall system performance, in terms of the number of transportation trips completed, compared to an otherwise identical system that lacks the aggression mechanism. Changing the behaviour 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 destructive interference is reduced.

Recently, two principled approaches to selecting an aggression level were shown to outperform the original random aggression function: the *global investment* [3] and *local investment* [9] methods were motivated by economic principles. *Local investment* was shown empirically to have the best performance, and was demonstrated to work on real robots.

Due to the entirely local nature of aggression interactions, where no agent has global system knowledge, suboptimal (from a global system performance point of view) solutions may be selected. One such suboptimal behaviour is mentioned (but not solved) in [3] and [9]: that of *worm–fights*, i.e. an aggressive interaction (or *fight*) between two robots, each at the head of a stable queue (or *worm*) of robots that are moving in the same direction. These worms are spontaneously occurring structures that can persist for long periods of time. Worms are beneficial to the overall system performance since the robots in a worm tend not to interfere with each other. In previous methods, including *local investment*, the outcome

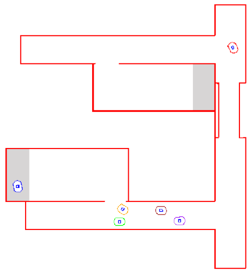

Fig. 1. The simulated *small world* used in the experiments. Populations of up to 24 robots were tested. This example scenario contains six robots.

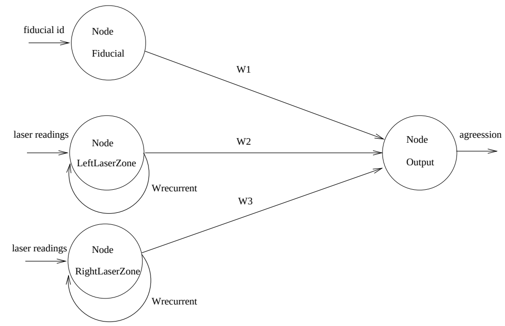

Fig. 3. The recurrent network mechanism used to calculate a robot's 'aggression level'.

of a *fight* was independent of the existence of worms, and fights tended, accidentally, to break up worms rather than preserve them. In this paper we present the novel *team local investment* aggression function, improving upon previous methods by enabling robots to win *fights* such that their dynamic worm/team structure tends to be maintained. We experimentally demonstrate that the new strategy provides a statistically significant improvement in system performance compared to *local investment* over a wide range of population sizes.

II. Rational Aggression

Consider a system of two teams of robots, Team Black and Team White, working in a narrow corridor as shown in Figure 2. They have the same task: transporting widgets from location A to location B at either end of the corridor. Assume that it is not practical for robots to transfer widgets between themselves. Team Black starts at A, and Team White at B. At some moment two teams block each other's progress inside a narrow corridor. Assume the robots have an internal aggression level, and can perform a stereotypical behaviour sequence called a *fight*, in which each robot displays its aggression to the other. If the leader of one team perceives that its rival has a higher aggression, it goes into a passive mode and can be pushed backwards, and consequently the robots that follows it retreat as well. Thus, *fight* resolves the spatial conflict by allowing the more aggressive team to push the other team backwards. From the point of view of system efficiency, which team should be more aggressive?

Suppose that Team Black has m robots, and Team White has n robots. The distances that the leaders of Team Black and Team White have travelled in the corridor are denoted as k and l respectively. The work spent on travelling inevitably generates real costs in terms of time, energy and computation. These are sunk costs which cannot be recovered. If we represent sunk cost in terms of distances that robots spend on moving back and forth due to losing a *fight*, the sunk cost caused by Team Black losing the *fight* is $k \times m \times 2$, $k \times m$ for the team to back up, and another $k \times m$ for the team to move back to the point where it previously meets Team White. Similarly, the sunk cost of Team White losing the *fight* is $l \times n \times 2$. *Local investment* determines robots' aggression based on the effort that an individual robot put into passing an area where interference between robots is more likely, for example, the narrow corridor in the figure. Therefore, it is a good candidate for selecting the winner when $k/l > n/m$, or $l/k > m/n$. On the other hand, if the corridor is short, ie. k/l is close to 1, and the population size of the system is large, m/n is more likely to be much larger than l/k; then, a strategy that is based on the internal state of single robot, such as local investment, becomes ineffective. In such a case, it may be more efficient to pick the leader of the team that has more robots to be the winner.

Therefore, we can break the aggression function into two parts depending on a threshold T. We pick $T = 3$ for this paper. The winner of a *fight* can be determined by the *local investment* scheme when neither side of the *fight* has more than T robots as it is more likely that either $k/l > n/m$ or $l/k > m/n$ holds true. Once the number of robots on either team reaches T, we set the aggression of the leader of that team to the maximum so that the worm wins the *fight* regardless of what aggression selection strategy its opponent is using (unless its opponent also has the maximum aggression).

III. Aggression Function Module

In contrast to previous aggressive robot systems [3], [8], [9] which used range sensors only, we assume the availability of a fiducial tracking sensor facing backwards on the top of each robot. Such a system can be implemented straightforwardly, and is available as a standard sensor in Player/Stage, the popular robot control and simulation system [5].

The fiducial sensor can determine the range, bearing and orientation of any fiducials (targets) within its field of view. Additionally, the sensor can read an integer value from each fiduciual: the fiducial id. We require that each robot displays a fiducial, and that the fiducial id can be dynamically changed. Initially, we initialize all of the fiducial ids to 1. Once with the aid of its fiducial sensor a robot detects that there are robots behind with the same orientation as its own, it takes a note of the fiducial id of the one that is the closest; then it adds 1 to that fiducial id and uses the sum as its own new fiducial id. Therefore, recursively, the fiducial id of a robot counts the number of robots behind it. The id of the robot at the head of a worm becomes an estimate of the worm length, ie. the number of robots in the worm.

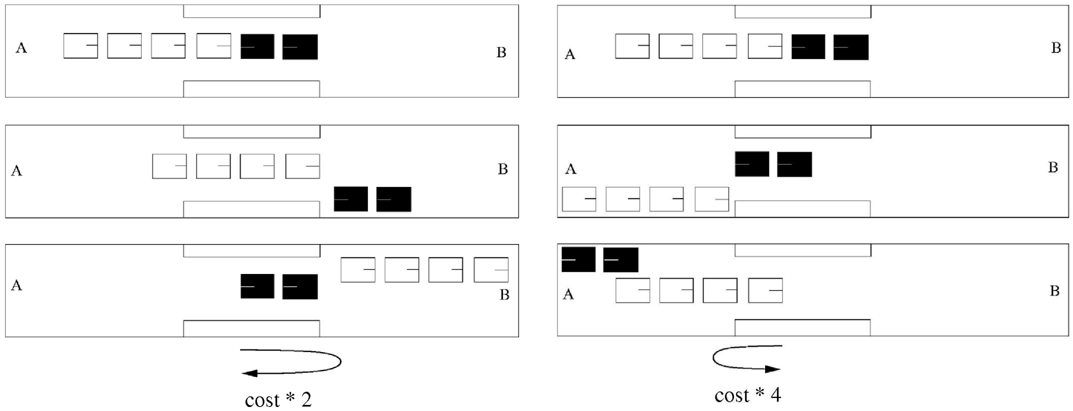

Fig. 2. Motivation for the *team local investment* strategy. The cost for each robot team to lose a *fight* is proportional to the sum of the cost for each robot, or approximately the cost for the head robot multiplied by the number of robots in the team. Whichever team has more to lose should win the *fight*: this is usually the larger team.

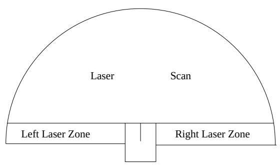

Fig. 4. The semicircle indicates the largest possible area covered by laser scan. The bottom left region of the semicircle is the left laser zone, and the bottom right region is the right laser zone. Thus, after a robot enters a narrow space, the areas of both of its left laser zone and right laser zone shall decrease.

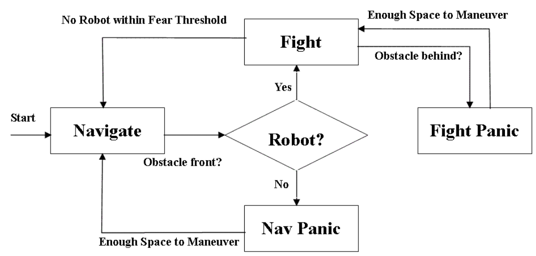

Fig. 5. Schematic of the robot control program.

In this paper, we use a recurrent network (RN), as shown in Figure 3, to implement our aggression function. A RN contains one or more nodes whose output is fed back into the network as input. RNs can be applied to reinforcement learning by allowing agents to retain memories of their previous states[2]. This model proved very convenient for implementing all the aggression functions we have studied. Our network contains two layers, an input–node layer and an output–node layer. The input–node layer has three input nodes. $Node_{llz}$ and $Node_{rlz}$ represents the open areas covered by the left laser zone (*llz*) and right laser zone (*rlz*), illustrated in Figure 4, each of the two nodes has a recurrent edge attached to it, which is necessary for implementing the *local investment* strategy because it enables a robot to memorize the distance it has travelled inside a narrow region. $Node_{fiducial}$ binds to the fiducial sensor readings. The output value of $Node_{output}$ is robot's 'aggression level'.

IV. Transportation Task

For direct comparison with aggression functions previously described in the literature, we use the identical experimental design and robot controller used in [3], [9]. The only change is in the aggression function employed. A team of robots operates in two simulated environments, the *small world*, Figure 1, and the *large world*, Figure 6. Robots, which initially start from a randomly selected location, must repeatedly traverse the environment between the shaded areas in both worlds. Since both worlds contain narrow corridors and doorways that allow only one robot to pass through at a time, robots constantly interfere with other. Besides the obvious size discrepancy, the major qualitative difference between two worlds is that the large world has two relatively long corridors while the small world has a short corridor.

A. Control Architecture

Each robot runs the same control program. Robots can be in one of three states: *Navigate*, *Panic* and *Fight*. Fig. 5 shows the permitted state transitions. *Navigate* guides the robot between goal locations by following walls and avoiding obstacles; it is the only goal–achieving behaviour. *Panic* and *Fight* are designed to cope with undesirable interference between robots. Time spent in *Panic* and *Fight* is unproductive overhead. Due to restricted space, we omit details of the controller which can be found in [3].

V. Experiment

A. Procedure

Three sets of experiments are performed in order to fully examine the performance, flexibility, and scalability of the *team local investment* strategy relative to other aggression functions. For each aggression function that we evaluate, we run 21 trials of experiments, and each trial is 7200 seconds long. In the first set of experiments, we compare two aggression functions, *local investment* and *team local investment*, using

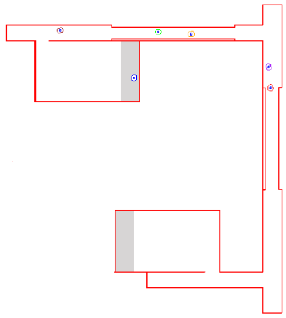

Fig. 6. The large world used in the experiments

six robots and the small world. In the second set, we run the team of six robots in the large world to determine whether the results of the first experiment are reproduced in a different work environment. In the last set, we examine the performance of both aggression functions and a non–fighting control in a range of population sizes, from one to twenty robots. Again, for each combination of aggression function, environment, and population size we run 21 experiments, each of 7200 seconds duration.

B. Measurements

Each robot logs the number of trips completed and the time spent in each of its three behaviours: *navigate*, *fight* and *panic*.

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

$$Trips_{team} = \sum_{i=1}^{n} Trips_{robot_i} \tag{1}$$

This value is easy to obtain and represents an objective measurement of the performance of the system as a whole. In terms of our resource transportation task, each trip completed by a robot is one unit of resource transported. Note that we are not trying to improve the number of trips that any single robot performs, but rather the total number of trips completed by all robots.

C. Simulation

Our experiments are done in simulation in Player/Stage. The simulated robots approximate the Pioneer–3DX at 0.44m long by 0.33m wide, with front and rear sonar rings and a laser range finder that approximates the SICK LMS-200.

D. Results

1) Experiment Set 1: By T–test, we show that the mean total resources transported by two aggression functions are statistically significantly different. Also, Table I reveals that the *team local investment* based aggression function performs significantly better than *local investment* in terms of the total number of resources transported per trial. The difference between two schemes can be seen in the histogram, Figure 7.

2) Experiment Set 2: Table II shows that in the large world *team local investment* again outperforms *local investment*. The T–test result, which is slightly above 0.05, and the histogram, Figure 8, show that the difference between the performances of two aggression functions is not as large as it is in the first set of experiments because the large world is more spacious than the small world, so when the number of robots running in the environment is small, the likelihood for robots to move together as a team is low. This is verified by the result of the next set of experiments, which shows that the difference between the performances of two aggression functions becomes more visible as the number of robots increases.

3) Experiment Set 3: Figure 9 shows the performance of three different types of controllers, *team local investment*, *local investment*, and *no fight* (ie. the controller with *fight* disabled), running in the small world as the population size increases. Robots barely interfere with each other when the population size is less than 3, so the performance of the three controllers increases linearly with respect to the population size. All three reach their maximum performance when the population size is 4 robots. As we add more robots, the curves diverge, and both *local investment* and *team local investment* outperform *no fight* by a large margin. Between the two aggression based controllers, the performance curve of local investment dips much faster than the curve of *team local investment*. Increasing population size elevates the probability for robots to form worms; thus, *team local investment*, which minimizes the costs of robot teams, becomes more effective than *local investment*, which minimizes the cost of individual robot only. The gap between two curves starts reducing at a population size of 14, and the two curves converge at 16. The bar chart in Figure 10 breaks down the fraction of time that robots spend in each behaviour at the corresponding population size for both aggression functions. In population sizes of 6 to 12, robots using *team local investment* spend more time navigating and less time panicking than robots with the *local investment* strategy. After the population size reaches 14, both aggression functions' behaviour decompositions are similar.

There are two reasons why *team local investment* loses its advantage over *local investment* when the number of robots reaches a certain level. First, as we add more robots, the world becomes too small for every robot to manoeuvre. Eventually when robots can not move anywhere without bumping into each other, these solutions become inadequate, as indicated by Figure 9. Second, as the population size increases, the likelihood of *fights* between two worms consisting more than 3 robots also increase. Since in our implementation of *team local investment* the leader of both worms have the maximum aggression, we see a performance hit because the *fight* between two robots with the same level of aggression usually takes longer to resolve than one between two robots with different aggressions.

Figure 11 shows the performance versus population size plots for two aggression functions with the large world. Again, initially the performance of each type of controller increases linearly with the population size when there is almost no

TrialType	Mean Resources	σ	N
Team Local Investment	285.86	11.42	21
Local Investment	270.19	14.61	21

TABLE I

SET 1 RESULTS: SUMMARY OF PERFORMANCE SCORES FROM TWO AGGRESSION FUNCTIONS BASED ON THE DATA COLLECTED FROM THE EXPERIMENTAL TRIALS WITH SIX ROBOTS IN THE SMALL WORLD.

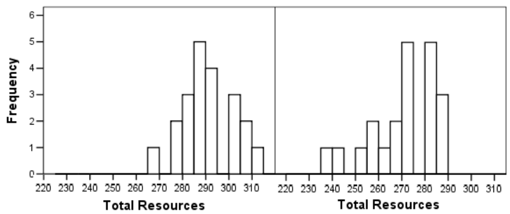

Fig. 7. Set 1 results: Histograms showing distribution of performance scores for two different controllers: *team local investment*(left), *local investment*(right)

TrialType	Mean Resources	σ	N
Team Local Investment	219.14	21.96	21
Local Investment	206.57	20.35	21

TABLE II

SET 2 RESULTS: SUMMARY OF PERFORMANCE SCORES FROM TWO AGGRESSION FUNCTIONS BASED ON THE DATA COLLECTED FROM THE EXPERIMENTAL TRIALS WITH SIX ROBOTS IN THE LARGE WORLD.

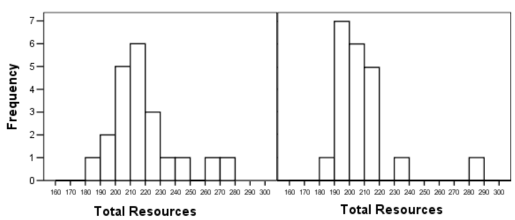

Fig. 8. Set 2 results: Histograms showing distribution of performance scores for two different controllers: *team local investment*(left), *local investment*(right).

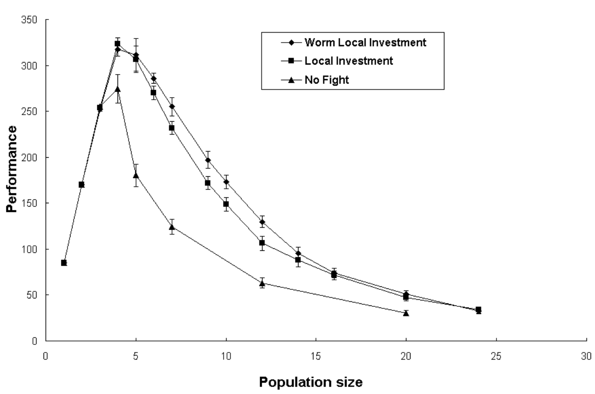

Fig. 9. Performances Vs. Population (Small World). Each data point is the sample mean (N=21) of the aggression function's performance at the corresponding population size. Error bars represent the 95% confidence interval for the mean.

interference between robots. *No fight* reaches its maximum performance when the population size is only 3 while two aggression based controllers produce much better maximum performances and are more adaptive to the expansion of population size. The gap between the curves of two aggression functions widens after they both hit the peak, but they converge as the population size reaches 14 (at which the error bars of the two curves overlap). Figure 12 compares the behaviour decompositions of the two aggression functions.

VI. DISCUSSION

A. Too many cooks revisited

One important contribution of this paper is to compare the effects of interference reduction techniques on scalability. To characterize multi-robot system performance, Vaughan proposed a hypothetical performance P versus population N curve (Figure 13 from [8]). We can call this the *cooks curve* (from the English proverb "Too many cooks spoil the broth"). Three benefits can be obtained by using an interference reduction technique:

1) increase the maximum value of P, labelled M in Fig. 13.
2) increase the area bounded by the cooks curve by improving performance P for some values of N.
3) expanding the range of values of N that produce acceptable values of P.

These are not exclusive: any combination may be seen. As shown in the Results section, our empirically obtained curves (Figs. 9,11) replicates this hypothetical model.

We can characterize the gain in system performance obtained by using aggressive interference reduction. Let us set our minimum acceptable performance level at three times the performance of one robot working alone. For both aggression functions, in both worlds, compared to non-aggressive robots, we see (i) increased maximum performance M; (ii) increased area under the curve; and (iii) increased width Q of the part of the curve with values of P greater than the allowed minimum.

B. Why use more robots than the optimum?

In these experiments, having too many robots reduces performance, and interference reduction is only useful when you have too many robots. Why would you ever have "too many" robots, with the associated cost? There are three main reasons. (i) *Accident:* the optimum number of robots is a complex function of the world, task, and robot design and is not necessarily predictable in advance. (ii) *Cohabitation:* if multiple groups of robots operate in the same environment the total population size may exceed the optimal population of any individual team. (iii) *Redundancy:* more robots give the system extra reliability. If we start our system with seven robots the performance is about 25% below the peak, but the system can endure the breakdown of up to three robots before the performance drops 20%. On the other hand, if we start with a population of four robots,the optimal population size, loss of two robots will drop performance by 50%. Thus, if preserving the stability of the system performance takes precedence over minimizing the operational costs, it will be better to start the

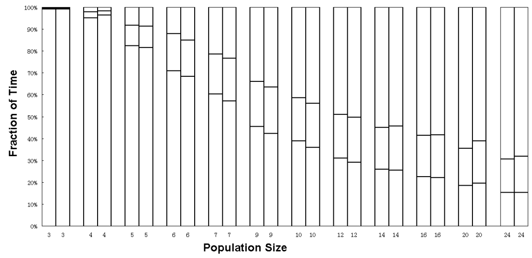

Fig. 10. Behaviour decomposition bar chart for small world data. The top, middle, and bottom stack of each bar represent the fraction of time spent in *panic*, *fight*, and *navigate*, respectively. Bars are grouped by the population size. The left bar of each pair accounts for the performance of *team local investment*, and the right bar accounts for *local investment*.

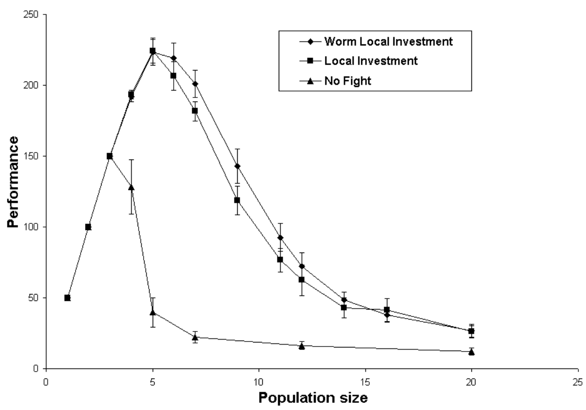

Fig. 11. Performances Vs. Population (Large World), presented as in Fig. 9

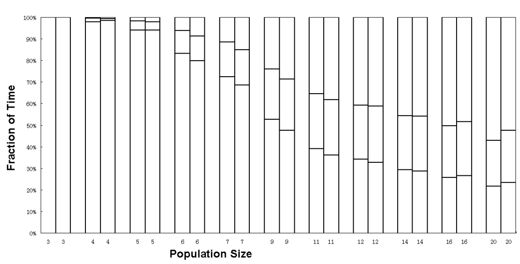

Fig. 12. Behaviour decomposition bar chart for large world data, presented as in Fig. 10

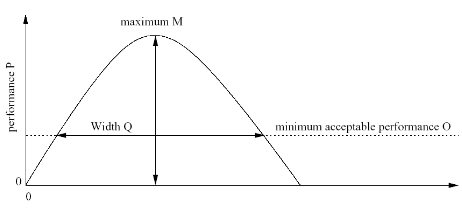

Fig. 13. Hypothetical correlation between performance and population size

system with a reasonably large number of robots even though it means that the system performance will be sacrificed initially.

C. Shortcoming of team local investment

The fixed threshold, $T = 3$, used by *team local investment*, loses its effectiveness as population increases and average worm length reaches 3. Fight between two worms consisting of more than 3 robots each have effectively random outcomes. To solve this issue, we may install another fiducial sensor, pointing forwards, on the robot; thus, the leaders of both worms will be able to read the length of the opposite worm. Then the worm that has less number of robots should yield the way to the longer worm.

VII. Future Work

A *fight* between two worms can be resolved more efficiently by adding another fiducial senor that points forward onto each robot. We expect this minor modification would give a significant boost to the performance of *team local investment*. Another valuable extension could be to devise adaptive aggression functions or parameters that optimize the system dynamically in changing environments.

VIII. Conclusion

We have described a new strategy for selecting robot's aggression, called *team local investment* which exploits and maintains the emergence of dynamic teams of robots. We presented experimental results that demonstrate that *team local investment* improves system performance over the best previously known method. The results show that utilizing aggressive robot technique expands the performance/population envelope of our multi–robot system: it improves peak performance and delays the reduction in performance with increased population size. These are the characteristics that define an effective interference–reduction technique.

References

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

[2] Bram Bakker. Reinforcement learning with long short-term memory. In *Advances in Neural Information Processing Systems 13 (NIPS' 13)*, 2002.

[3] Sarah Brown, Mauricio Zuluaga, Yinan Zhang, and Richard T. Vaughan. Rational aggressive behaviour reduces interference in a mobile robot team. In *Proc. of 12th Int. Conf. on Advanced Robotics, Seattle, WA, USA*, 2005.

[4] Jacques Ferber. *Multi-agent systems : an introduction to distributed artificial intelligence*. Harlow : Addison-Wesley, 1999.

[5] 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 Int. Conf. on Intelligent Robots and Systems*, pages 1226 – 1231, 2001.

[6] 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.

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

[8] 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. of Int. Conf. Simulation of Adaptive Behaviour, Paris, France*, 2000.

[9] Mauricio Zuluaga and Richard T. Vaughan. Reducing spatial interference in robot teams by local-investment aggression. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Edmonton, Alberta, Canada*, 2005.