Distributed Formation Control of Heterogeneous Robots with Limited Information

Michael de Denus, John Anderson, and Jacky Baltes

Autonomous Agents Laboratory,
Dept. of Computer Science, University of Manitoba,
Winnipeg, Manitoba, Canada R3T2N2
{mdedenus,andersj,jacky}@cs.umanitoba.ca
http://aalab.cs.umanitoba.ca

Abstract. In many multi-robot tasks, it is advantageous for robots to assemble into formations. In many of these applications, it is useful for the robots to have differing capabilities (i.e., be heterogeneous) in terms of perception and locomotion abilities. In real world settings, groups of robots may also have only imperfect or partially-known information about one another as well. Together, heterogeneity and imperfect knowledge provide significant challenges to creating and maintaining formations. This paper describes a method for formation control that allows heterogeneous robots with limited information (no known population size, shared coordinates, or predefined relationships) to dynamically assemble into formation, merge smaller formations together, and correct errors that may arise in the formation. Using a simulation, we have shown our approach to be scalable and robust against robot failure.

1 Introduction

Formations are desirable in collections of robots for many reasons, including maximizing area coverage, sensor coverage, and minimizing contact while moving quickly as a group. Heterogeneity in robots is similarly advantageous in many settings: parsimony in control, budgetary considerations, and specialization of tasks all make simpler robots with divergent capabilities desirable. Knowledge may be inconsistent between robots, making for more diversity, and commonly-assumed global knowledge may not be available. For example, in a rescue scenario, robots may be lost due to failure, and agencies may arrive with new equipment over time, preventing total population from being known with certainty. Communication may be noisy or temporarily disrupted, leading to inconsistency in information across members. All of these factors conspire to present a difficult challenge for the creation of formations and their maintenance over time.

In this paper, we present a distributed approach to formation control in multi-robot teams that deals with such challenging scenarios: robots may be heterogeneous in terms of movement abilities, sensing, and other equipment; knowledge of others, including identity and the overall size of the team, is assumed to be

S. Behnke et al. (Eds.): RoboCup 2013, LNAI 8371, pp. 292–303, 2014.

imperfect; and communication is assumed to be unreliable. Our approach is also intended to assume that failure of robot hardware is a strong likelihood over time, and that other robots may be available to replace them by physically sensing or encountering the ongoing formation in the environment.

Our approach deals with these challenges by limiting the number of robots that each formation member interacts with directly, while not putting any restriction on the size of the team itself. This theme has been successful in the past [1,2]. Existing methods, however, do not address addition and removal of robots from the formation, and are not particularly tolerant of robot failure. We have created a mechanism for new robots to gain entry to the formation, and for existing robots to adapt given the failure of others. The combination of these two properties addresses the fact that we do not know how many other robots are in the environment, and that the robot population can change at any time. This builds on our previous work [3], which assumed only that communication had to be local (no broadcast) and where heterogeneity in sensing was limited to the inability to identify other robots.

2 Related Work

There are many existing approaches to formation control. One common theme is to employ elements of central processing. Some approaches restrict processing to a single robot [4], while others pre-compute portions of the paths centrally [5]. Any central processing introduces a single point of failure, and should be avoided in cases when possible. Under our approach, the responsibility for formation control is distributed, and no single robot is essential.

Another common theme is to use leader and follower robots. One way to accomplish this is through hierarchical leadership [2,6]. These methods define multiple levels of leadership, but still suffer from the fact that the failure of any of the leaders would disable some portion of the formation. Our approach would be affected by the loss of a robot, but has mechanisms to recover from such a loss. Techniques involving chains of robots can also be considered to fall under this category. Such techniques can either have a predefined ordering of robots [1] or can determine this ordering at runtime [3]. Our technique employs the concept of following a target, but does not have any predefined relationships.

Another large class of formation control techniques that use local rules to maintain formations in a somewhat decentralized way. For example, Hadaegh et al's approach [7] allows for the dynamic addition and removal of formation members, but does not scale well, as all members need to communicate with one another. Leonard and Fiorelli [8] demonstrate a method that employs virtual robots, whose purpose is to direct and organize the formation. This approach is robust against robot failure, but requires robots to have a shared frame of reference. Other approaches of this type [9,10] are able to maintain formations, but are very domain specific and do not allow large numbers of potential formations. Our approach allows for a large number of potential formations, and requires communication only with a robot's immediate neighbors.

3 Approach

Our approach to formation control employs several strategies to deal with heterogeneity and limited information. To create a formation in a distributed manner, formations must be built up as robots encounter one another in the environment. Procedures must be in place for smaller sub-formations to merge, and for the formation to continue to emerge even as robots fail. As groups of robots join into larger formations, the formation itself must be continually balanced, so that a globally coherent shape emerges (e.g. if forming a V, smaller formations either cannot all join one side, or must be repositioned if they do). Each of these operations is detailed in the following subsections, following some preliminary discussion to explain common elements of our approach. Because the approach is distributed, it must also be possible for *any* robot in the formation to perform any required operation as needed. Under our approach, robots can initiate direct communication with other robots that have been perceived or have been previously identified, but can not perform broadcast communication.

3.1 Preliminaries

Our approach falls into the *neighbor referenced* category of Balch and Arkin's categorization [11]. Each robot in the formation has one associated target robot that it maintains a position with respect to. This target robot is determined dynamically, and is used to position the robot within the formation. It is also possible for a robot in a formation to follow no one in some specifications (e.g. the point of a V).

A formation in our approach is specified as a collection of interconnected *segments*, each consisting of chains of robots formed over time. Each segment has an associated angle and distance, specifying the desired angle and distance at which each robot in the segment should keep its target. These may be specified as functions, allowing segments to be lines or curves. Each segment also has a relative length, which defines proportion in the formation by specifying the desired length of the segment relative to the length of the shortest segment in the formation's definition. Overall, this allows a broad range of specified formations, including open shapes (lines, Vs) and closed shapes (squares, rectangles) as well as formations with curved segments. The only current restriction on shape in our implementation is that an agent can have only one target to follow.

Any number of segments may connect together, and these connection points are termed *entry points*. The robots occupying these positions in the formation are *entry point robots*. These robots are responsible for handling joining and merging operations, as well as estimating population size. These entry point robots are not pre-determined: any robot becomes one if it happens to be located at the point where a segment ends. A robot knows this by examining the segment membership of its neighbours, and the number of neighbouring robots.

Each possible formation is common knowledge among all participating robots, as is the desired current formation out of the many the robots may possibly know, and a goal direction in which to move.

To deal with an unknown population size, each entry point robot maintains an estimated count of each segment it belongs to. These in turn can be used to estimate the size of an entire formation. To maintain a segment population count, entry point robots send out *counting* messages to each of their neighbours. Upon receiving a counting message, if the receiver occupies an entry point, it increments the count in the message by 1, and sends it back to the sender. A non-entry point robot will have at most two neighbours, and so a non-entry point-robot receiving a counting message increments the count by one forwards it on to the neighbor who did not send it. If messages are lost, or the count becomes stale, it will be corrected by a future message.

To obtain an estimated count of the entire formation, a robot examines all segments to which it belongs. It sums up the relative lengths of these segments, and calculates the percentage of the total formation in these segments. It then uses this information to extrapolate the number of robots in the complete formation. This must be considered a rough estimate, since it is making the assumption that a robot's current neighborhood is representative of the formation as a whole. However, it strikes a balance between accuracy and the extensive communication beyond immediate neighbours that forcing entry point robots to exchange population counts would entail.

3.2 Joining a Formation

There are two ways in which a robot can join a formation: creating a new formation with another robot, or joining an existing formation. In our approach, both of these options are initiated in the same way. When not participating in a formation, robots periodically send out *join* requests to those they encounter. A robot receiving such a request (the *responding* robot) may be in three different situations. First, the responding robot may not yet be in a formation, in which case it would reply with a *success* message, along with instructions to the sender for joining the new formation. Second, the responding robot may be in a formation but not at an entry point, in which case the responding robot would reply with a *rejection* message. If the responding robot has a target, it will identify that target to the requesting robot as an alternative. Finally, in the case where this is a valid entry point, the responding robot can reply with a success message and integrate the joining robot. To perform this integration, the entry point robot will choose the destination segment that is most out of balance and adopt the requestor as a neighbor there (this is a common element to balancing an existing formation, and is explained in Sec. 3.4). An example of successfully joining a formation is illustrated in Fig. 1. Loss of communication during joining will cause the joining robot to maintain its heading and speed while trying to locate another place to join.

3.3 Merging Formations

In our approach, merging two distinct formations forces robots in one formation to sequentially join the other, and thus makes use of the techniques described in

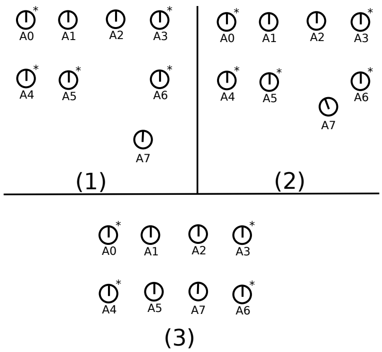

Fig. 1. The joining process. The formation pictured is a rectangular box formation. Entry points are marked with '*' (1) Robot A7 detects a nearby formation, initiating communication with A6. (2) A6, being an entry point, directs A7 towards its left side. (3) A7 Joins the formation. A5 recognizes that it is now part of a longer segment and drops entry point status.

Sec. 3.2. When any robot perceives another, it queries that robot for formation membership, and if it is part of a formation, requests a size estimate. At the same time it performs a size estimate of its own formation (Sec. 3.1). The smaller of the two formations is asked to transform into a line formation (this involves a chain of adopting one's nearest neighbour, rejecting anyone further, until everyone has only a single neighbour), to facilitate joining the larger formation one at a time. The robot at the front of the new line has an entry point to merge to, located in the other formation. This causes a chain of formation joining, as each robot in the line recognizes it has no target and attempts to join the formation next to it.

3.4 Balancing a Formation

Because robots may join at any entry point in the formation, balance needs to be maintained between segments and ultimately across the formation. To accomplish this, we employ a balancing technique that allows the entry point robots to move other robots between segments. The entry point robot first computes the ideal number of robots in each of the segments in which it participates. This is done by estimating the total number of robots in these segments (Sec. 3.1), and then performing a vector multiplication according to the fractions that should be in each segment according to the desired proportion for each segment, specified in the formation description. The resulting values are called the *ideal* counts for these segments. The segment whose actual count is furthest below its ideal count is selected as the destination segment, and the segment which is furthest above its ideal count is selected as the source segment. Assuming these differences are at least one, the entry point agent instructs its neighbor in the source segment to become its neighbour in the destination segment, and its neighbour in the destination segment to become the neighbour to the shifted agent (Fig. 2). While this moves only one robot at a time, it is important to view this occurring in a distributed manner: if a distant part of the formation is short on robots, multiple local shifts will eventually cause them to move there.

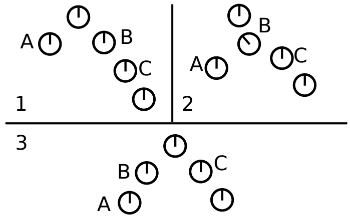

Fig. 2. In panel 1, the V formation is unbalanced. In panel 2, the entry point robot has instructed robot B to change segments. In panel 3, we see the result of the rearrangement of robots into the new balanced formation.

3.5 Communication Interruptions

All the above operations require ongoing communication, but only across small local distances. Still, these may be interrupted at times. In general, the worst case for communication interruption during any of these operations results in a breaking up of a formation (a heartbeat not heard after ten seconds in our implementation leads one robot to think its neighbour has failed). In that worst case, the robots are still close to one another and can pick one another up again as communication resumes, through the operations defined in the previous subsections.

Inconsistencies are also possible because of communication problems. For example, because it is possible for communications to break down in the middle of a balance, it is possible for an entry point robot to ask its neighbor to shift to a different segment, and yet have the neighbor occupying that position to remain where it is, resulting in two agents occupying the same position in the formation. This is repaired by periodically checking the intended segment ID for each of an entry robot's neighbours with that actually reported by the neighbour, and managing conflicts by requesting one robot to move, essentially restarting the rebalance. There are a number of other analogous measures for dealing with potential inconsistencies; all of these may be found in [12].

4 Implementation

We implemented this approach using the well-known player/stage [13] simulation package, using Pioneer2 DX robots equipped with laser and fiducial scanners, with both of these sensors operating in a forward direction. We used three different sensing ranges, allowing for heterogeneity in sensing. We also used three separate maximum speeds, giving heterogeneity in locomotion ability.

Our robot controller is behaviour-based [14], and consists of three behaviours. These behaviors are the same as those used in our previous work [3]: *goal seeking*, *formation keeping* and *neighbor avoidance*. Goal seeking causes all robots to move towards a known common goal location. This is used to simulate localization and goal selection. Formation keeping ensures that robots stay at the desired distance and angle from their target robot. Finally, neighbor avoidance ensures that robots don't get too close together or hit each other.

Because some robots will have short sensing distances, we allow them to be assisted by those with better sensing abilities, by querying their immediate neighbors for the visibility of other robots. This allows robots to get a reasonable idea of whether a specific robot is in front of or behind them. In practice, this type of message is usually sent to a robot's target. The target then returns whether or not it can sense the sending robot. If a robot is visible to its target while the target faces forward, this indicates it has passed the target and should be behind it. In such a case, the following robot slows down and allows the target to resume its appropriate position.

Heterogeneity in locomotion ability is addressed by having each robot send out messages to its neighbors. These messages indicate the minimum of the robot's top speed, and the top speed of its neighbors. Eventually, this will propagate the speed of the slowest member to all formation members. This allows all formation members to properly maintain formation without leaving members behind.

Each robot has a common representation of the formation in terms of relationships between segments. Segment membership is stored as a bit vector in each robot.

5 Evaluation

Because the main thrust of this approach is its support of heterogeneity, we examined the effects of heterogeneity on scalability and tolerance of failure. Since other approaches do not support the range of conditions for which ours is intended, a direct comparison was not possible.

As a basis for performance, we defined a measure of formation error. Based on the total number of robots in the formation, we calculate the ideal number of robots per segment. We then calculate the absolute difference between the number of robots in the segment, and this ideal. The formation error is the sum of these absolute differences from the ideal segment counts. Other metrics employed include the maximum number of robots in formation, and the number of trials that result in a single formation. Further metrics and additional experimentation can be found in [12].

The trials were run in an obstacle-free simulated environment, which was approximately 40 metres wide by 1000 metres long. Robots assembled into a V formation with two segments of equal length. Robots were required to maintain a 45 degree angle, and maintain a separation of 2 metres. Robots were initially distributed across random grid points within a 6 metre x 6 metre area within the environment. Further dispersion of robots from this is observed initially, because of obstacle avoidance and desire to move toward the goal.

We defined three different levels of perceptual and locomotion ability, and we refer to these levels as *poor*, *moderate* and *good*. For sensing, we used a range of 6 metres for poor, 8 metres for moderate, and 10 metres for good. For locomotion ability, we used 1 m/sec for poor, 2 m/sec for moderate, and 3 m/sec for good. We established 7 profiles, each describing the percentage of robots in the simulation that have each level of ability along each of these dimension. These profiles are described in Table 1. In the case when the number of robots in a trial is not evenly divisible into the specified groups, remaining robots were granted moderate sensing or locomotion.

Table 1. Sensing and locomotion profiles

Profile Number	Poor	Moderate	Good
0	0%	100%	0%
1	25%	75%	0%
2	50%	50%	0%
3	0%	75%	25%
4	0%	50%	50%
5	25%	50%	25%
6	50%	0%	50%

To examine scalability, we varied the number of robots in the simulation. We used populations of 5, 10 and 15 robots. For population, we tested all 7 locomotion and all 7 sensing profiles. To examine robustness against failure, we used 10 robots, and selected 3 chances of failure, each paired with a maximum number of failed robots. Every 30 seconds, our logging server tells a robot to fail with a given probability. The probabilities of failure were 0, 0.1 and 0.4. These were paired with a maximum number of 0, 1 and 2 robot failures respectively. As with the scalability trials, we tested all 7 sensing and locomotion profiles within each category of failure.

We ran our trials on Amazon EC2, using 50 c1.medium instances, performing 50 repetitions for each trial. A trial was considered complete when either 5 minutes had elapsed, or all robots were in the same formation with no robot changing segments for 10 seconds.

6 Results

This section provides an overview of our findings. A more detailed description, along with many additional results, can be found in [12]

Our results showed that neither heterogeneity in sensing, nor heterogeneity in locomotion ability had a large impact on robots' ability to create and maintain formations using our approach. In Figs. 3 and 4, we see that the maximum number of robots in formation at a given time is high for all sensing profiles. This indicates that formations are being created, and maintained in different areas in the environment.

Despite a high percentage of robots in formation, ending a trial by achieving a single overall formation was not common ($\sim 5\%$ of trials). Part of the reason for this is the distance between robots given the distributed nature of this approach. Once robots have coalesced into only a few sub-formations, it can become difficult for robots from different formations to encounter one another and merge further. While a single formation may not always be the end goal, we intend to experiment further with an additional behaviour attracting robots to others perceived at a distance, and other related behaviours that may influence the likelihood of a single end formation.

The results in Fig. 5 and 6 show that the formations that are being created are generally correct. The large error bars are present because this statistic is sampled at the end of the trial. If a correct formation is not achieved, the formations may be in the middle of a joining or balancing operation. This potential for increased error means that this is actually quite a conservative metric.

The results referenced in this section are for the variation of sensing profiles only. Variation of locomotion profiles had a similar effect. Full results can be found in [12].

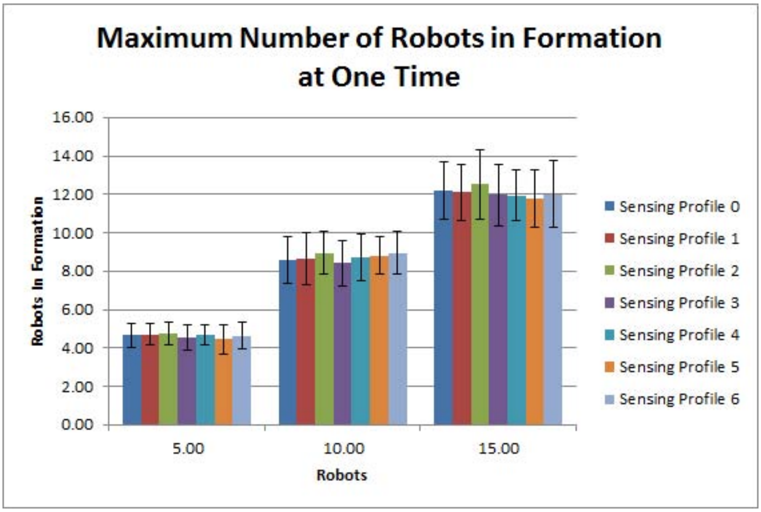

Fig. 3. Maximum number of robots in formation as population size changes

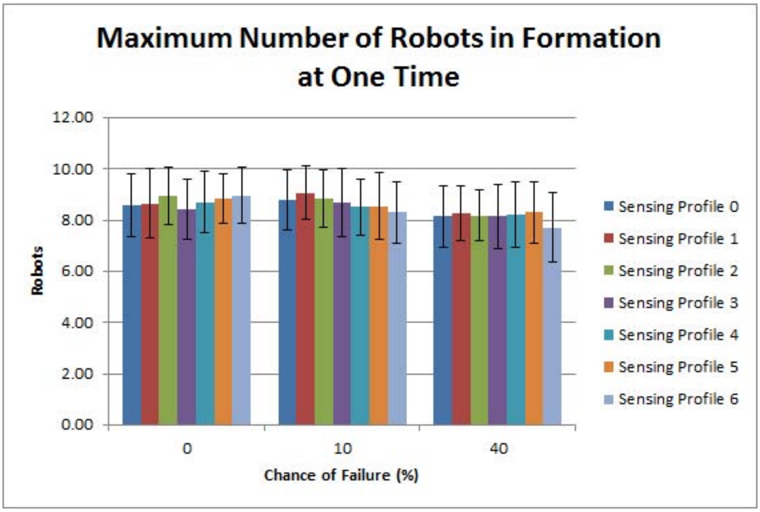

Fig. 4. Maximum number of robots in formation as chance of failure changes

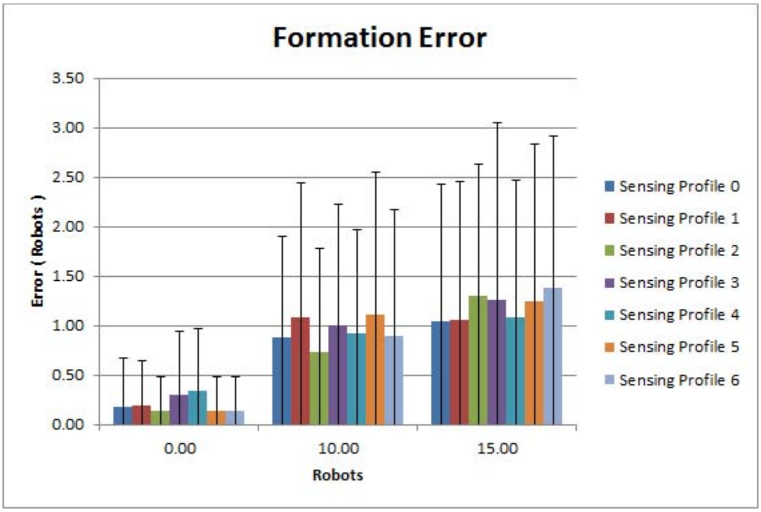

Fig. 5. Formation error as population size changes

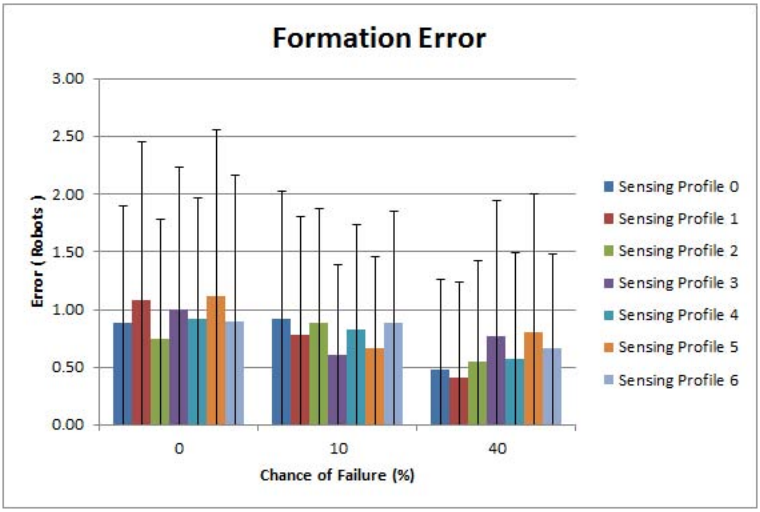

Fig. 6. Formation error as chance of robot failure changes

7 Discussion and Future Work

We have described a distributed approach to formation control that is independent of the number of robots and supports heterogeneity in sensing and locomotion. Using a simulation, we have shown that our approach is robust against robot failure, and that it is scalable across a number of different population sizes.

These are positive results given the significant challenges of the conditions under which this operates. We demonstrated that neither heterogeneity in sensing nor heterogeneity in locomotion ability seriously impact the performance of this approach. The establishment of formations also indicates that our approach adequately compensated for limited knowledge of others and the inability to assume a given perceptual or locomotion ability.

There are a number of immediate avenues of future work. First, we would like to experiment with even larger populations of robots, and extend heterogeneity to elements such as robot size and terrain that may not be traversable by all robots. In addition, we would like to experiment with more complex formations. We also plan to explore metrics for evaluating formations that better reflect decentralized approaches such as this. We would like to implement several existing approaches to formation control, and use these metrics to get a better idea of how our approach compares to others.

We are currently moving to replicate these experiments using Citizen microrobots in a mixed reality environment using global vision [15], after the prior RoboCup Mixed Reality League. The small size of the microrobots allows a reasonable population on a small physical area, while the virtual elements employed in mixed reality can allow a large surface by having terrain that scrolls under the robots, as well as obstacles represented virtually. Since catastrophic obstacles (e.g. fire, pits) can be simulated, would also allow a rationale for robot failure to replace the statistical failure used in the simulation studies.

References

1. Fredslund, J., Mataric, M.: A general algorithm for robot formations using local sensing and minimal communication. IEEE Transactions on Robotics and Automation 18(5), 837–846 (2002)
2. Kwon, J.W., Chwa, D.: Hierarchical formation control based on a vector field method for wheeled mobile robots. IEEE Transactions on Robotics 28(6), 1335–1345 (2012)
3. de Denus, M., Anderson, J., Baltes, J.: Heuristic formation control in multi-robot systems using local communication and limited identification. In: Baltes, J., Lagoudakis, M.G., Naruse, T., Ghidary, S.S. (eds.) RoboCup 2009. LNCS (LNAI), vol. 5949, pp. 437–448. Springer, Heidelberg (2010)
4. Hattenberger, G., Lacroix, S., Alami, R.: Formation flight: Evaluation of autonomous configuration control algorithms. In: Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Jose, USA, pp. 2628–2633 (October 2007)
5. Rampinelli, V., Brandao, A., Martins, F., Sarcinelli-Filho, M., Carelli, R.: A multi-layer control scheme for multi-robot formations with obstacle avoidance. In: International Conference on Advanced Robotics, ICAR 2009, pp. 1–6 (June 2009)
6. Consolini, L., Morbidi, F., Prattichizzo, D., Tosques, M.: Stabilization of a hierarchical formation of unicycle robots with velocity and curvature constraints. IEEE Transactions on Robotics 25(5), 1176–1184 (2009)
7. Hadaegh, R., Kang, B., Scharf, D.: Rule-based formation estimation of distributed spacecraft. In: The 10th IEEE International Conference on Fuzzy Systems 2001, vol. 3, pp. 1331–1336 (2001)
8. Leonard, N., Fiorelli, E.: Virtual leaders, artificial potentials and coordinated control of groups. In: Proceedings of the 40th IEEE Conference on Decision and Control 2001, vol. 3, pp. 2968–2973 (2001)
9. Lee, G., Chong, N.Y.: Adaptive self-configurable robot swarms based on local interactions. In: Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Jose, USA, pp. 4182–4187 (October 2007)
10. Yamaguchi, H.: Adaptive formation control for distributed autonomous mobile robot groups. In: Proceedings of the 1997 IEEE International Conference on Robotics and Automation, vol. 3, pp. 2300–2305 (April 1997)
11. Balch, T., Arkin, R.: Behavior-based formation control for multirobot teams. IEEE Transactions on Robotics and Automation 14, 926–939 (1998)
12. de Denus, M.: Adaptive formation control for heterogeneous robots with limited information. Master's thesis, University of Manitoba (2013)
13. Gerkey, B., Vaughan, R., Stoy, K., Howard, A., Sukhatme, G., Mataric, M.: Most valuable player: a robot device server for distributed control. In: Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, pp. 1226–1231 (November 2001)
14. Arkin, R.C.: Behavior-based Robotics. MIT Press, Cambridge (1998)
15. Anderson, J., Baltes, J., Tu, K.Y.: Improving robotics competitions for real-world evaluation of AI. In: Proceedings of the AAAI Spring Symposium on Experimental Design for Real-World Systems, Stanford, CA, pp. 1–8 (March 2009)