Collective Construction with Multiple Robots

Jens Wawerla Gaurav S. Sukhatme Maja J. Matarić

Robotics Research Laboratory, Computer Science Department
University of Southern California
Los Angeles, CA 90089-0781, USA
jens@robotics.usc.edu, mataric@usc.edu, gaurav@usc.edu

Abstract

We study the problem of construction by autonomous mobile robots focusing on the coordination strategy employed by the robots to solve a simple construction problem efficiently. In particular, we address the problem of constructing a linear 2D structure in a planar, bounded environment. A "minimalist" single-robot solution to the problem is given, as well as two multi-robot solutions, which are natural extensions to the single-robot approach, with varying degrees of inter-robot communication. Results show that with minimal inter-robot communication (1 bit of state), there is a significant improvement in the system performance. This improvement is invariant with respect to the size of the environment.

1 Introduction

Construction is a vast industry worldwide. Large-scale construction already relies on operator-assisted automation to a significant extent. However, applications remain where human presence is undesirable, prohibitively expensive, or impossible. Examples include construction and repair in hazardous areas after natural disasters such as earthquakes, construction under extreme physical conditions such as high pressure (undersea), extreme temperature or the lack of oxygen (undersea, space), and remote areas which are not readily accessible to humans and typically require initial infrastructure to make human arrival easier or even possible (Martian surface). In such environments, construction by autonomous robots might be the only viable alternative.

Motivated by these applications, we study the problem of construction by autonomous mobile robots. Our focus is on the *coordination strategy* employed by the robots to solve a simple construction problem efficiently. In particular, we address the problem of constructing a simple 2D structure in a planar, bounded environment. We exhibit a "minimalist" single-robot solution to the problem as well as two multi-robot solutions, which are natural extensions to the single-robot approach, with varying degrees of inter-robot communication. It is worth noting that our work does not specifically address one of the am-

bitious domains mentioned above; instead we focus on a fundamental question - how should inter-robot communication be structured in order to improve the time efficiency of cooperative construction?

In our experiments, the robots are required to build a continuous, linear barrier out of octagonal cardboard blocks. The blocks are equipped with either positive or negative Velcro and the robots can distinguish them by the color of the block. The robots have to attach blocks of alternating colors in order to make them stick together.

Using trials with a single physical robot we validate that our algorithm can solve this task and build a barrier. Then, we show that multiple robots can solve the same task and perform better if they locally communicate the color of the last block in the barrier to each other. The multi-robot trials are performed in simulation, but use the same underlying controller as the physical robot.

The remainder of this paper gives brief overview of related work, followed by a description of our controller. The section on experimental work presents the single robot and multi-robot experiments, in which two approaches (with and without communication) are compared in environments of varying size.

2 Related Work

Previous research on automated construction and wall-building can be classified using various attributes. These include: 1. the main emphasis of the research (mechanical design vs. algorithmic issues); 2. the testbed and approach (simulation vs. hardware implementation); 3. the nature of the robots (manipulators, mobile robots, hybrid combinations); and 4. the size of the robot team.

In [14], the authors report a system for automated assembly using artificial potential functions. This (simulation-based) work is focused on sequencing complex construction tasks and does not address the problems of object manipulation. In [6] the authors show a rule-based minimalist approach to collective sorting. In [9], a defensive wall-building system is reported, with a set of simple robots called the *U-bots*. These minimalist approaches

are strongly biologically-inspired and rely heavily, in contrast to our approach, on probabilistic foundations. Such methods are not usually time-efficient and depend on frequent disassembly of incorrectly built parts. On the other hand, the *U-bots* require only a very small number of simple rules. In other related work, an approach for a climbing and walking robot that builds 3D structures out of polymer foam [2] focused largely on the mechanical design of the robot and not on the control issues. In [3], a neurocontroller for groups of agents is described. This simulation-based work focuses mainly on neurally-controlled agents rather than robots with perception and actuation uncertainties. In [11] a vehicle for a masonry robot is reported, and in [10] a theoretical design for a mobile masonry robot is proposed. While their work does not focus on multi-robot construction, [1, 12] present algorithms and simulated systems of stigmergic construction with large-scale swarm agents. Besides actual construction, foraging is an important part of this work. In [5] several foraging approaches, using interference among the robots, are compared.

The work mentioned above [1, 3, 12] is purely simulation-based, focuses on the manipulators executing CAD plans, or uses time-consuming probabilistic models. In contrast, our work described in this paper focuses on local control to build a given structure with autonomous mobile robots.

3 The Construction Task and Controller

3.1 The Construction Task

To focus on the coordination problem and to keep different approaches comparable, the construction task we studied is fairly simple. The robots are required to build an approximately straight, continuous barrier out of octagonal cardboard blocks. Continuity here means that the barrier has no visible openings, i.e., the blocks are contiguous. For experimental convenience, the blocks are lined with positive or negative Velcro, so if attached in the correct order they stick together. To enable a robot to distinguish between the two Velcro sides, the blocks are colored differently depending on the type of Velcro on them. The starting point and the orientation of the barrier are indicated with a laser beacon [7], and the barrier is 'seeded' with an initial block placed under the beacon.

3.2 The Controller

We use a behavior-based controller [8]. Its three main components are: virtual sensors which provide high-level sensor information, behaviors which encapsulate tasks the robot can perform, and actuators, which abstract and handle the low-level control of the robot. The actual robot hardware is accessed using Player¹, a device server that

provides a powerful, flexible interface to a variety of sensors and actuators [4].

Task sequencing and reactive control are encoded in an inhibition network among the virtual sensors and behaviors, as shown in Figure 1. Each robot is equipped with the following behaviors and virtual sensors:

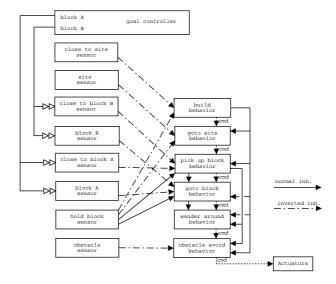


Figure 1: The Controller

Obstacle Sensor continuously observes the surroundings of the robot and provides distance information about obstacles.

Block Sensor provides the (x, y) position, relative to the robot, of a block in the field of view of the camera.

Site Sensor detects the laser beacon and provides the robot with the following information: distance from robot to beacon, robot bearing relative to the center of the beacon, and the orientation of the beacon relative to the robot.

Hold Block Sensor tells the robot whether it is holding a block with its gripper.

Obstacle Avoidance Behavior prevents the robot from bumping into obstacles, by slowing it down if it gets close to an obstacle, and causing it to back up if it gets too close to an obstacle.

Wander Around Behavior causes the robot to drive straight. If the robot is not holding a block it pans the camera after several seconds in order to look for a block. Random wandering emerges from combining this behavior with the obstacle avoidance behavior. The robot travels straight until it comes close to an obstacle, and changes its heading in response. After successfully avoiding the obstacle, the robot has a new heading and switches back to the wandering behavior.

Goto Block Behavior moves the robot to a block detected by the block sensor.

Pickup Behavior picks up a block with the gripper.

¹Player was developed jointly at the USC Robotics Research Labs and HRL Labs and is freely available under the GNU General Public License from http://playerstage.sourceforge.net.

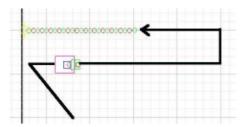


Figure 2: Finding the attachment point using only local perception

Goto Site Behavior moves the robot to the construction site detected by the site sensor. In the multi-robot case, when close to the site, a robot asks for permission to add its block to the barrier. If no other robot answers within a certain time, the robot starts attaching its block.

Build Behavior causes the robot to add a block to the end of the barrier. To do so the robot orients itself according to the laser beacon, follows the already built portion of the barrier, performs a 90° turn, looks for the front of the barrier in the camera image, turns 90° again, and approaches the end of the barrier (see figure 2). If the color of the last block in the barrier is identical to the color of the block the robot is carrying, it turns away and drops the block elsewhere. Otherwise it attaches the block to the barrier. During the execution of this behavior the robot responds to every *permission to build* message with a *site occupied* message, thereby denying other robots access to the construction site.

Goal Controller stores the color of the block the robot was processing last. Since robots do not globally communicate their location, their status etc., nor do they know their relative position to the construction site, there is no way to predict what block color is needed next. Therefore, the goal controller alternates the block color the robot is working on. By disabling virtual senors for one block color, the robot focuses only on the other block color, as appropriate.

4 Experimental Design

To validate the approach, we performed two series of experiments. In the first series, we used a single ActivMedia Pioneer robot to show that the controller described above successfully builds a barrier in a real-world environment with significant sensor and actuator noise. In the second series of experiments we used a simulation to investigate the performance of the same controller in a multi-robot case and explore the benefit of local communication between robots.

4.1 Robot Experiments

We used an ActivMedia Pioneer DX2 robot equipped with a color camera, a gripper, a ring of 16 ultrasound sensors, a SICK laser range finder and wireless Ethernet.

The environment we used is rectangular, $7m \times 5m$ in size, and surrounded by a 40 cm high wall. A laser beacon

was mounted in the middle of one of the longer walls of the arena. The robot could read the beacon quite reliably from almost any point in the arena as long as it faced the beacon with the laser range finder. A green cardboard block was placed in front of the beacon so the robot could gain all the information about the barrier it is supposed to build from the environment. The start point and orientation of the barrier are provided by the beacon, and the sequence of blocks is determined by the 'seed' cardboard block underneath the beacon. An equal number of red and green cardboard blocks (10 total) were randomly distributed in the arena.

At the beginning of the experiment the robot was randomly placed in the arena without any knowledge about its own location, the position of the cardboard blocks, the layout of the arena, or the location of the laser beacon. The only information given to the robot *a priori* was the color of the initial cardboard block underneath the laser beacon.

4.2 Simulation Experiments

We performed the multi-robot experiments in simulation only, largely due to limitations of hardware availability and reliability.

The simulation trials were performed in *Stage*, a high-fidelity multi-robot simulator [13]. Stage² simulates all the sensors and actuators that were used in this work and also allows simulated object manipulation. Since Stage also uses the *Player* [4] device server, the controller used on the Pioneer robot was used in simulation without any major changes.

Two environments were used for the simulation experiments. The small environment was $14m \times 14m$ and the large environment $28m \times 28m$ (four times as large). Other than size, these environments were identical. Each is a rectangle surrounded by a wall that is detectable by the sonars as well as the laser range finder, as in the real robot case. A laser beacon was located in the middle of one of the walls and a green block was placed in front of it. In addition, sixteen blocks (8 red and 8 green) were alternately placed along the wall opposite the beacon and along one of the other remaining walls. The initial position of the robots was along the walls with the blocks, approximately 5m from the blocks and facing them.

In all simulation experiments the robots had no knowledge about the environment, the initial block number or location, their own location, or the location of the beacon. Since the robots did not know the color of the initial block, they randomly chose one when they were powered up. After the initial choice each robot alternated the block color every time it set a block down.

²Stage was developed jointly at the USC Robotics Research Labs and HRL Labs and is freely available under the GNU General Public License from http://playerstage.sourceforge.net.

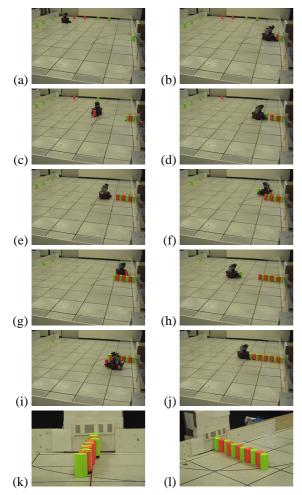


Figure 3: Building a barrier of 10 blocks with a single robot

5 Experimental Results

5.1 Robot Experiments

The task assigned to the robot was to build a barrier by attaching 10 blocks to the initial block placed near the beacon. We defined success as the creation of a continuous, straight, 10-block barrier. Since we are interested in multi-robot coordination rather than precise object manipulation, and since the Pioneer sensor and actuator suite does not allow high precision, we used completion time as our main performance metric.

Over three trials, barrier construction times were almost identical: 30.5, 29.9 and 30.6 minutes. Although few, these validation experiments demonstrate the functionality of the controller in a physical environment and provide a foundation for the simulation.

Figure 3 shows one of the barriers that was built by the physical robot. Figure 3(k) shows the front view. The first 3 blocks underneath the laser beacon are oriented very precisely, the following ones have a slight offset in their position due to inaccuracy in finding the end-point of the barrier using the color blob information from the

Table 1: Results: 'no memory about the last block'

Robots	1	2	4	6	8
$\mu_{time} [sec]$	2971	2730	2221	1600	2507 ³
σ_{time}	312	560	611	605	1314
μ_{failed}	0.6	5.9	9.8	7.5	8.9
σ_{failed}	0.5	2.4	4.8	3.6	4.1

camera. Figure 3(1) gives an overview of the barrier and shows that the barrier is continuous. Note that the lines on the floor are not used by the robot. The position and orientation of the laser beacon (and therefore of the barrier) was intentionally chosen to match the pattern on the floor in order to provide a reference for a human analyst. Figure 3 shows a sequence of images taken during an experimental trial.

5.2 Simulation Experiments

After validating that the controller works on a real robot, the remaining (multi-robot) experiments were performed in simulation.

The "No Memory" Case. In this set of simulations the robots did not know the color of the last block in the barrier. Therefore, they started the *build behavior* whenever they reached the construction site, which meant they might have had to ultimately drop the block, because of its incorrect color.

The experiment was performed with 1, 2, 4, 6 and 8 robots. Each set was simulated 10 times. To prevent extremely long simulation times, experiment time was limited to 4500 seconds.

The results are shown in Table 1 and in Figure 4. It can be seen that the mean success time μ_{time} decreases with an increased number of robots. Furthermore the results show the saturation effect typical in multi-robot systems. In our domain, interference mainly happened in the waiting zone around the construction site. Under optimal conditions this zone can hold seven robots. It appears that the performance peak is between six and eight robots.

Minimalist Communication. In the second set of experiments, we extended the controller from the "nomemory" case set by adding minimalist communication, as follows. The robot that performs the *build behavior* checks for the last block in the barrier before attaching a new block, in order to guarantee that the blocks would stick together. This may result in either attaching the block or turning away and dropping the block. In either case the robot knows the color of the last block in the barrier and broadcasts this information. Other robots that are in the waiting zone around the laser beacon receive this message and decide to either keep waiting (since their block is of the wrong color) or ask for permission to en-

³Simulation time limit was exceeded twice, so the barrier was incomplete with 7 and 8 blocks.

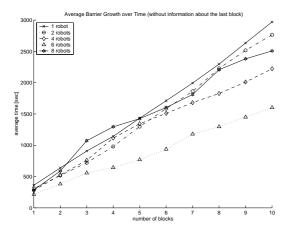


Figure 4: Barrier growth over time: the 'no memory' approach

ter the construction corridor. Robots that are not waiting but approach the waiting zone shortly thereafter make the same decision. This technique gives the robots knowledge about the block sequence in the barrier.

For small group sizes, it is possible that the majority of the robots can be in the waiting zone with a block that is not currently needed. This situation lowers the performance of the system and may result in a deadlock. To avoid that, we implemented a frustration level that grew with the amount of waiting time. If a robot's frustration exceeds a preset threshold, it turns away from the construction site and drops the block at a safe distance from the laser beacon. After that it proceeds to look for and pick up a block of the other color.

In situations in which more than one robot with the required block is waiting, conflicts may occur about which robot should go first. These are solved in the following manner. After receiving the information about the last block in the barrier, each robot with the correct block waits for a random time before requesting permission to enter the corridor. In addition, the time period during which a robot requests permission is randomly chosen (remember the robot assumes it has permission to enter the corridor if no other robot denies its request). This makes conflicts highly unlikely.

The results are shown in Table 2 and Figure 5. The average success time decreases with the number of robots and saturates with eight robots. As the small number of failed blocks due to timeout $\mu_{timeOut}$ shows, the negative impact (of waiting) on the system is very small. Even with this additional error, the system performs better than the *no memory* approach. Due to the high variance, the 95% t-test significance is only approximately 60%. Figure 6 compares the approaches. Since the time for foraging and construction is relatively constant, the number of failed blocks actually determines the performance of the system. With a probability of 99%, the second approach produces fewer failed blocks compared to the first.

Table 2: Results: 'Minimalist communication'

Robots	2	4	6	8
$\mu_{time} [sec]$	2552	1788	1397	1964 ⁴
σ_{time}	471	398	363	1408
$\mu_{wrongType}$	4.5	3.1	1.4	2.0
$\sigma_{wrongType}$	2.2	2.3	1.6	2.1
$\mu_{timeOut}$	0	0.5	1.0	1.6
$\sigma_{timeOut}$	0	0.5	0.9	2.3
μ_{all}	4.5	3.6	2.4	3.6
σ_{all}	2.22	2.4	2.1	3.8

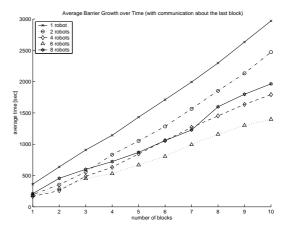


Figure 5: Barrier growth over time: minimalist communication

Small vs. Large Environment. By changing the size of the environment, we were able to change two characteristics: 1) the ratio of construction to foraging time, and 2) the ratio between the number of robots to the available space. In both cases a larger environment meant less interference among the robots due to more space and relatively shorter waiting time at the construction site.

The experiments described in the previous section were also performed in a larger environment $(28m \times 28m)$. In this case the maximum simulation time was set to 5500 sec. Tables 3 and 4 show the results. Comparing the no memory with the minimalist communication approach in the large environment shows similar results as seen in the smaller environment. The performance increases with the number of robots, and communicating the color of the last block locally decreases completion time by limiting the number of failed blocks. As a result of the larger environment, the robots have more space and interference is decreased, so the performance does not drop in the eight robot case. When comparing the results from the small and the large environment, comparing time has little meaning since it heavily depends on the size of the environment. Comparing the average number of failed blocks, on the other hand, shows that the environment size does not have any influence on performance.

⁴Simulation time limit was exceeded twice, so the barrier was incomplete with 7 and 9 blocks.

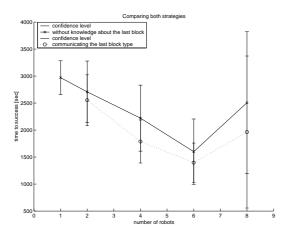


Figure 6: Comparing the average success times

Table 3: Results: Large environment: 'no memory about the last block'

Robots	4	6	8
$\mu_{time} [sec]$	4236	3312	2672
σ_{time}	539	954	321
$\mu_{wrongType}$	9.1	8.2	7.9
$\sigma_{wrongType}$	3.9	2.3	3.7

6 Conclusion

This work addressed one aspect of the very large field of robotic construction. Specifically, we focused on the multi-robot coordination problem in cooperative construction. We presented a behavior-based robot controller for a construction task, and demonstrated that it could build a barrier of octagonal cardboard blocks with an accuracy limited only by the precision of the robot's actuators and sensors. We presented experimental results with multiple, simulated robots, cooperating to build a barrier in the same environment, and compared the effects of communication and of two different environment sizes.

Our simulation experiments show that increasing the number of robots improves the performance of the system. The barrier in our experiments is built using only two kinds of blocks (distinguished by color). Robots must coordinate to ensure that alternate colored blocks are placed next to each other, in order to guarantee a stable wall. Our experiments show that locally communicating one bit of state (the color of the last attached block) improves the performance of the system since it limits the number of 'failed' blocks. By running the same experiments in a larger environment we showed that such 'minimalist' communication works even if the robots spent more time foraging than waiting or building.

The problem of reducing interference around the construction site dominates the overall construction problem. In future work we plan to address this further by sequenc-

Table 4: Results: Large environment: 'Minimalist communication'

Robots	4	6	8
$\mu_{time} [sec]$	3569	2696	2667
σ_{time}	544	431	348
$\mu_{wrongType}$	5.9	3.9	4.1
$\sigma_{wrongType}$	2.5	1.3	2.2
$\mu_{timeOut}$	0.2	0	0.6
$\sigma_{timeOut}$	0.4	0	0.5
μ_{all}	6.1	3.9	4.5
σ_{all}	2.6	1.3	2.1

ing the blocks according to their color while the robots take them to the construction site. We also plan to investigate multiple construction points for a single structure.

Acknowledgments

This work is supported in part by DARPA (grant DABT63-99-0015) under the MARS program and by ONR (grant N00014-00-1-0638) under the DURIP program.

References

- Eric Bonabeau, Marco Dorigo, and Guy Théraulaz. Swarm Intelligence -From Natural to Artificial Systems. Oxford University Press Inc., 1999.
- [2] Adrian Bowyer. Automated construction using co-operating biomimetic robots. Technical report, University of Bath Department of Mechanical Engineering, November 2000.
- [3] F. Crabbe and M. Dyer. Second-order networks for wall-building agents. In International Joint Conference on Neural Networks, volume 3, pages 2178– 2183, Washington D.C., July 1999.
- [4] Brian P. Gerkey, Richard T. Vaughan, Kasper Støy, Andrew Howard, Gaurav S. Sukhatme, and Maja J Matarić. Most valuable player: A robot device server for distributed control. In Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS), pages 1226–1231, Wailea, Hawaii, October 2001.
- [5] Dani Goldberg and Maja J. Matarić. Interference as a tool for designing and evaluating multi-robot controllers. In *Proc. of the Natl. Conf. on Artificial Intelligence (AAAI)*, pages 637–642, Providence, Rhode Island, July 1997.
- [6] Owen E. Holland and Chris Melhuish. Stigmergy, self-organisation, and sorting in collective robotics. Artificial Life, 5(2):173–202, 1999.
- [7] Andrew Howard, Maja J. Matarić, and Gaurav S. Sukhatme. Relaxation on a mesh: a formalism for generalized localization. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 1055–1060, Wailea, Hawaii, October 2001.
- [8] Maja J Matarić. Behavior-based control: Examples from navigation, learning, and group behavior. *Journal of Experimental and Theoretical Artifical Intelligence*, 9(2–3):323–336, 1997.
- [9] Chris Melhuish, Jasnon Welsby, and Charles Edwards. Using templates for defensive wall building with autonomous mobile ant-like robots. In Proc. of TIMR, Towards Intelligent Mobile Robots, Bristol, UK, March 1999.
- [10] G. Pritschow, M. Dalacker, J. Kurz, and J. Zeiher. A mobile robot for on-site construction of masonry. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, volume 3, pages 1701–1707, September 1994.
- [11] H. F. Steffani, J. Fliedner, and R. Gapp. A vehicle for a mobile masonry robot. In Proc. of the IEEE Intl. Conf. on Industrial Electronics, Controll and Instrumentation (IECON), volume 3, pages 1337–1342, 1997.
- [12] Guy Théraulaz and Eric Bonabeau. Modelling the collective building of complex architectures in social insects with lattice swarms. *J. Theor. Biologie*, 177:381–400, 1995.
- [13] Richard T. Vaughan. Stage: A multiple robot simulator. Technical Report IRIS-00-393, Institute for Robotics and Intelligent Systems, School of Engineering, University of Southern California, December 2000.
- [14] Louis Whitcomb, Daniel Koditschek, and Joao Cabrera. Toward the automatic constrol of robot assembly tasks via potential functions: The case of 2-d sphere assemblies. In *IEEE International Conference on Robotics an Automation*, pages 2186–2190, Nice, France, May 1992.