Cooperative Multi-Robot Box-Pushing

Maja J Matarić **Martin Nilsson** **Kristian T. Simsarian**

Volen Center for Complex Systems
Computer Science Department
Brandeis University
Waltham, Ma 02254
tel: (617) 736-2708 fax: 736-2741
maja@cs.brandeis.edu

RWCP* Novel Functions SICS Laboratory
POB 1263
S-164 28 KISTA, Sweden
fax: +46 (8) 751-7230
tel: +46 (8) 752-1570
mn@sics.se kristian@sics.se

Abstract

This paper deals with communication in task-sharing between two autonomous six-legged robots equipped with object and goal sensing, and a repertoire of contact and light-following behaviors. The performance of pushing an elongated box toward a goal region is difficult for a single robot and improves significantly when performed cooperatively, but requires careful coordination between the robots. We present and experimentally demonstrate an approach that utilizes cooperation at three levels: sensing, action, and control, and takes advantage of a simple communication protocol to compensate for the robots' noisy and uncertain sensing.

1 Introduction

This paper is concerned with the role and effect of communication in mobile robot systems that require task-sharing. We focus on manipulation tasks with sufficiently challenging dynamics to require the careful cooperation of two or more robots. Sensing and actuation is noisy and uncertain in mobile robot domains, resulting in partial knowledge about the world. We explore the use of communication to speed up performance on a manipulation task that is possible with a single robot but becomes more efficient through cooperation. We describe a simple communication protocol that compensates for the robots' incomplete and inaccurate information about the environment while also reducing interference and resource competition between the robots.

We chose box-pushing as the problem domain because it has both theoretical interest and practical applications as it is an instance in a large class of practical object manipulation tasks that appear to require tight feedback and control of real-world physics and dynamics. From a theoretical standpoint, box-pushing is a variant on the canonical object manipulation problem that draws on issues in fine motion as well as high level planning and control. Box-pushing is related to the well-known "piano-movers problem" (Schwartz & Sharir 1983), in that it requires the achievement of the top-level goal of delivering the box to a particular location, as well as the maintenance of low-level requirements including obstacle avoidance, maintaining contact with the box, and maintaining forward motion. From a practical point of view, box-pushing is a prototypical problem for studying various tasks requiring cooperation of a number of smaller robots moving larger objects.

Our domain is set up so that the box-pushing task does not require cooperation, but is shown to be significantly improved by it because we are interested in evaluating the benefits of cooperative solutions relative to single-agent approaches. In our scenario, the dynamics of the robot and box interaction are designed so as to increase efficiency in terms of forward movement of the box, and communication and turn-taking is needed to satisfy both the high-level and the low-level goals of the system.

We experimentally demonstrate that our box-pushing scenarios falls into the class of problems in which communication can produce superlinear performance improvements on problems that otherwise allow only linear speed-up with cooperative solutions (Huberman 1990). We give strong evidence that single-robot and non communicating two-robot solutions are inferior, in terms of performance, compared to the communication and turn-taking protocol given in this paper.

2 Experimental Environment

We used two commercially available Genghis-II six-legged robots in our experiments (Figure 1). The robots are fully autonomous with on-board power

---

*Real World Computing Partnership

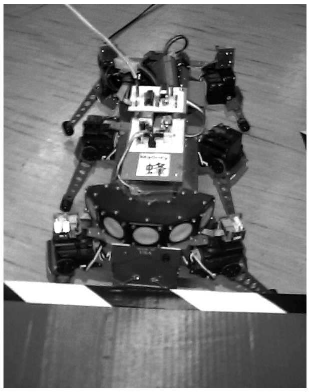

Figure 1: Two Genghis-II six–legged robots were used in the experiments. Each is 30 cm long and 15 cm high, equipped with two whiskers, five pyroelectric sensors, and the communication hardware, mounted on top. In the shown experiments, the two robots are connected with a 5-meter communication cable.

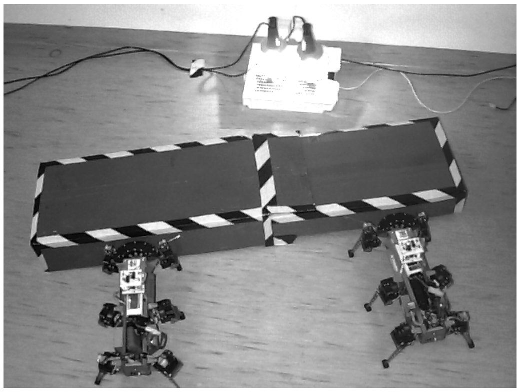

Figure 2: The experimental environment consists of the two Genghises, the box, and the goal region, marked with the lights. The box is low enough to allow the pushing robots to see the lights, and long enough to require force on both sides in order to move straight.

and computation, programmed in the Behavior Language based on the Subsumption Architecture (Brooks 1986). The dynamics of legged robot pushing are less well understood than those of more common wheeled locomotion. When perpendicular to the box, the robots "push" it with their whiskers. However, at more acute angles of incident, they "kick" the box with the closest leg, which turns it faster than would a pushing force exerted by a flat or round robot. The cooperative algorithm must account for the concurrent and independent displacement of the box by each of the robots.

We introduced communication facilities by adding wires directly to unused pins of the main processor to implement an I^2C interface with a bit rate of about 20 kbit/s, limited by the speed of the main on-board 68HC11 processor. The interface was connected to our customized board mounted on top of the robot, and it communicated with the control program via primitive operations in the Behavior Language, requiring no changes to the system software. The communication board has two main responsibilities. One is to allow us to add arbitrary sensing and actuating extensions, which can be controlled by the user program running on the main processor, via the I^2C interface. The other is to handle the communication with other robots to which one of the two PIC16C84 processors on the board is dedicated.

The communication system runs completely asynchronously with the main processor and is designed for radio communication. We expect to have the radios fully operational in the near future but in the experiments described here, we used a 5-meter cable between the robots that could easily transmit 50 kbit/s without errors. Higher speeds are possible with the existing hardware, but would not be meaningful due to the limitations of processor speed and radio bandwidth.

The two robots operate in a square arena, approximately 3 meters to a side. The box to be pushed is 14 cm high, 35 cm wide, and 1.3 meters long. The goal region, located at one end of the arena, is indicated by two tungsten light sources that emit IR radiation detectable by the robot's pyroelectric sensors (Figure 2). In the experiments described here, the goal location was fixed in order to make the trials comparable, but it can be can be moved before and during experiments.

Two sensing modalities were employed for this task: the whiskers and the pyros. The left and right whiskers each provide binary contact information which is combined in a tight control loop that servos the robot's body so as to keep it in contact on both sides and thus perpendicular to the box. We will call this the *correcting* behavior. If only one of the sensors is contacting the box, the robot turns in that direction in order to reestablish full contact. If contact is lost on both sides, the robot moves in the direction of last contact (one bit of memory is kept).

The semi-circle of five pyroelectric sensors detects a moving heat edge. The sensing algorithm we used computes the absolute difference of light responses with respect to time and saves the maximum direction. The *light-following* behavior turns the robot in the direction of the maximum light; it turns the robot to the left if either of the leftmost two are the highest, to the right if either of the rightmost two are, and moves it straight forward otherwise.

If *correcting* and *light-following* are in conflict, the former is given higher priority. The *pushing* behavior is the result of *correcting*. The robots are not equipped with sufficient sensors to distinguish different objects in the environment, and will push anything that contacts their whiskers. In our environment, the box is the only such object. However, in a more complex environment with obstacles, other simple sensors can be easily added to the system in order to add object discrimination capabilities.

3 The Approach

The interaction dynamics of two six-legged walking robots with the environment, each other, and concurrently manipulable box, are complex. They are made more so due to the noise, error, and uncertainly in the sensors and effectors. Consequently, the designer is left with a choice of either explicitly programming in a partial model of the system, or implicitly embedding one in the controller. Our work addresses the latter approach.

Using an implicit model for control effects the extent to which each agent models the other(s) in order to effectively interact. Since maintaining explicit models of the other agent's internal and external state and goals has a computational, sensory, and communication overhead (Gasser & Huhns 1989, Rosenschein & Genesereth 1985), our previous work on collective robot behavior (Matarić 1994, Matarić 1992) demonstrated that certain effective group behaviors can be achieved without explicit models. This work follows in the same vein, exploring simple cooperative strategies that utilize communication to minimize the need for explicit modeling and prediction.

In our scenario, the robots have no explicit knowledge of the world physics, of their own mechanics and dynamics, nor of those of the other robot. Furthermore, they do not make explicit predictions about the effects of their or the other's actions. Instead, their control strategy is reactive, based on immediate sensory information including the communication channel with the other robot. Our underlying assumption is that the inter-robot and object interactions are Markovian, resulting in a scenario in which history plays no significant role: the state of the system at any point in time can be completely determined by the inputs into the robots' sensors. Uncertainty and asynchrony can challenge this assumption, so we imposed a synchronized turn-taking protocol keeps the robots' sensory and communication information more correct and current. In contrast, if the robots were to move concurrently, they would be unable, in our system, to communicate the necessary sensory data in order to remain reactive and maintain an effective pushing strategy.

The following is the basic control loop, identical for both robots, consisting of two main concurrent processes. The robots have a set of four primary actions, $A = \{Stop, Go - forward, Left, Right\}$, After performing any action, the robot performs the *correcting* behavior to assure its contact with the box.

```
Whenever "my-turn" message is received:
    get own sensory data $s$
    get other-robot's sensory data $o$
    combine $s$ and $o$ into current joint state $j$
    use $j$ to select the best next action $a$ from $A$
    use $a$ to select the best message $m$ from $A$
    send $m$
    perform $a$
    perform correcting
    send "your-turn" message

Whenever an action-message $m$ is received:
    perform $m$
    perform correcting
```

The outlined algorithm embodies a task-sharing strategy in that the robots take turns controlling the actions at each time step. This contrasts a centralized framework in which both robots would be directed from a global controller and also from a master-slave scenario in which one of the robots would retain control. Our two robots are completely equal in terms of their control capabilities and the amount of information they have about the world. Each depends on the other for the state of the "other side" of the box, but when it obtains that information, each can proceed independently it if its turn. The system utilizes cooperation at three levels: sensing, action, and control. Not only do the robots serve as each other's remote sensors and effectors, but the robustness of the pair is increased because the control system is duplicated and independent across the two robots. Consequently, any transient errors on either of the robots will be at least partially compensated for by the other, as long as both are functioning. If either fails completely, the system is reduced to a single robot, and its performance drops significantly, as is described in the next section.

4 Experiments and Results

We conducted three sets of experiments. In the first set, a single robot used a simple reactive strategy to push the box toward the goal. In the second, two robots used identical simple strategies without synchronization. Finally, in the third, the two robots used a cooperative communication and synchronization strategy.

In all experiments the goal and starting locations were the same. Two starting locations were used for the box, one on either side of the workspace. The robots were initiated from positions almost in contact with the box. To record their positions over time, a grid was laid down on the floor and a pause was introduced after each turn, during which the positions of the robots were hand-plotted. Due to the pause, the time is recorded in 6-second steps rather than continuously. The resolution of the plots is approximately 15cm or to 5% of the space. Tests were not done to check the effect of this pause on the task performance but all experiments were carried out this way and their positions plotted in the same manner to provide an even comparison. The total number of runs for all positions is 38. The figures show a scaled version of the robots and the environment, and plot typical paths taken in the experiments. The data is obtained from the position information, and hand-plotted. The starting and ending configurations for the box and robot are shown, the path is dotted, and the goal is indicated with a circle.

The first set of experiments was used to evaluate the difficulty of the task for a single robot. The robot was started away from the box, facing it, and walking forward. When it found the box it proceeded to push it toward the goal using the *correcting* and *light-following* behaviors. Data were gathered from two different starting points of the box, and also of the robot along the box, in the middle (Figure 3), and on the outside. From the middle starting point, without a more complex control algorithm, the robot was unable to maneuver the box toward the light. From the outside starting position, the box was positioned so that it was guaranteed to reach the goal, and the system did so in 15.5 steps on the average (Figure 4). This set of experiments was performed as the control, in order to compare the single-robot and two-robot performance.

The box could be reliably pushed toward the goal by a single robot if it alternated sides. In the ideal case this strategy would take twice as long as when the box is pushed by two turn-taking robots, one at each end. Due to the legged locomotion employed here, the time to perform a strategy which alternates pushing sides would effectively be more than double because the robot does not have great dexterity in performing lateral movements.

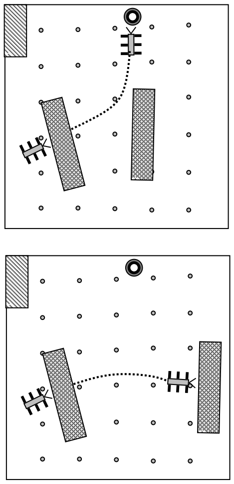

Figure 3: This figure shows the two typical manifested behaviors from a number of runs when a single robot is started at the box center. The robot either maintains contact with the box and loses track of the light, or abandons the box for the light.

The second set of experiments tested a efficacy of multi-robot non-cooperative strategy. The experiments consisted of testing the performance of two completely independent box-pushing robots, each positioned and initiated at one end of the box. It might seem that the two should, while simply executing *correcting* and *light-following*, deliver the box to the goal. Instead, we found that the actions of one robot frequently undid progress made by the other, resulting in two failure modes: pushing the box out of bounds "out of bounds" (Figure 5) or abandoning it. The “out of bounds” region is the area where the robots, if pushing the box, cannot sense the goal, and thus cannot recover. Furthermore, unlike the single-robot case, no starting point could be found where the robots would be guaranteed success. In the small number of fortuitously successful trials, the average number of steps to reach the goal was 8.

In the final set of experiments, the robots were given the turn-taking communicating control strategy described in the previous section. In all runs from both starting positions the box was successfully maneuvered to the goal (Figure 6). The turn-taking min-

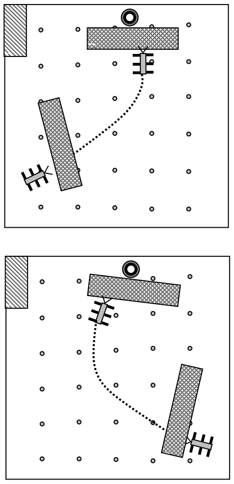

Figure 4: This experiment was performed as a control case. The robot was positioned so as to guarantee reaching the goal with the box. The average number of steps taken was 15.5.

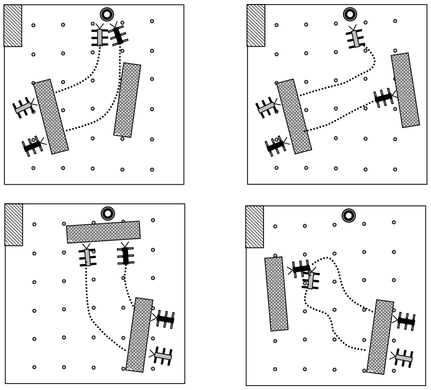

Figure 5: Typical trials of two non-communicating box-pushing robots. In this case the robots could not synchronize and used different behaviors. From left to right, in the first case both robots abandoned the box for the light. In the second, one robot abandoned the box for the light while the other pushed it out of the arena. In the third case, the robot successfully pushed the box together, while in the fourth case the outside robot abandoned the box and interfered with the progress of the inside robot. In successful trials, the average number of steps to reach the goal was 8.

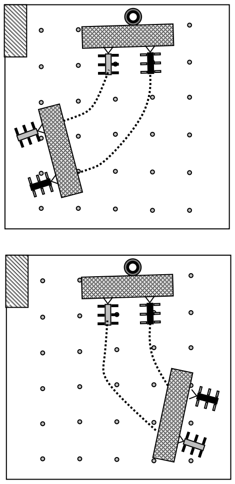

Figure 6: In this case the robots were synchronized and communicating, and were successful, in all cases, in pushing the box toward the goal by cooperatively performing the turning and walking behaviors. The average number of steps to reach the goal was 6.

imized undoing of each other's work, and the communication expanded the robots' perceptual space, resulting in a much more effective performance. The average number of steps to successfully reach the goal was 6, which is less than 50% the time required by a single robot (15.5 steps), and 25% less time than required by a pair of independent robots (8 steps).

5 Related Work

While there is a large body of work on abstract and simulated multi-agent systems, very few physical multi-robot systems have been implemented. This section reviews only those systems that have dealt with cooperative object manipulation tasks. Our work differs in two aspects. Mechanically, our experiments are performed on legged robots, whose pushing dynamics are novel and have not yet been studied extensively. Computationally, our approach is minimalistic both in terms of the required robot behaviors and the communication strategy.

The systems that are most related to ours are those designed to be cooperative. In these systems, the two or more robots are aware of each other's existence, and can sense and recognize each other directly or through communication. In contrast to our work, Caloud, Choi, Latombe, LePape & Yim (1990) and Noreils (1993) remain faithful to the state-based framework, and apply a planner-based control architecture to a box-moving task implemented with two large wheeled robots in a master-slave configuration. At the other end of the control spectrum, Kube (1992) describes a series of simulations of wheeled robots performing a collection of simple reactive behaviors, including locating and pushing a box, that are being incrementally transferred to physical robots. Donald, Jennings & Rus (1993), and their related work, give a theoretical grounding for information requirements in performing various geometric tasks. They introduce a transformational approach for deriving equivalent robot protocols using different resources, and experimentally demonstrate their work on a pushing task with two wheeled sofa-moving robots. Our approach is complementary in that in our system communication compensates for the robots' sensory limitations, while in theirs the sensory capabilities are shown to eliminate the need for explicit communication. Finally, using an approach most similar to ours, Parker (1994) describes a behavior-based task-sharing architecture for heterogeneous robots, and demonstrates it on a successful box-pushing experiment using communication between a wheeled and a six-legged robot.

6 Summary and Conclusions

This paper has addressed the problem of cooperative box-pushing with two autonomous six-legged robots. The chosen task was achievable by a single robot but could be performed much more effectively with two robots. We demonstrated a simple cooperative strategy that greatly outperforms both a single-robot alternative and an approach with two non-communicating robots. The strategy uses cooperation at the level of sensors, effectors, and control; it uses turn-taking to achieve implicit coordination and maximize the accuracy of the sensors, uses simple communication to expand the robots' sensory capabilities, and uses physical redundancy to increase robustness of control.

One of the motivations for this project was to set up a basis for a system in which the robots would learn to communicate, instead of having a hand-coded protocol. In a set of experiments described in Matarić & Simsarian (1995), we extended the described system to one in which the robots learn, in real time and from a small number of trials, what to communicate to the other in order to effectively deliver the box to the goal. We are optimistic about extensions of this and other simple approaches to cooperation and communication in complex multi-robot domains, and continue to pursue them.

References

Brooks, R. A. (1986), 'A Robust Layered Control System for a Mobile Robot', *IEEE Journal of Robotics and Automation* **RA-2**, 14–23.

Caloud, P., Choi, W., Latombe, J., LePape, C. & Yim, M. (1990), Indoor Automation with Many Mobile Robots, *in* 'IROS-90', Tsuchiura, Japan, pp. 67–72.

Donald, B. R., Jennings, J. & Rus, D. (1993), Information Invariants for Cooperating Autonomous Mobile Robots, *in* 'Proc. International Symposium on Robotics Research', Hidden Valley, PA.

Gasser, L. & Huhns, M. N. (1989), *Distributed Artificial Intelligence*, Pitman, London.

Huberman, B. A. (1990), 'The Performance of Cooperative Processes', *Physica D* **42**, 38–47.

Kube, C. R. (1992), Collective Robotic Intelligence: A Control Theory for Robot Populations, Master's thesis, University of Alberta.

Matarić, M. J. (1992), Minimizing Complexity in Controlling a Collection of Mobile Robots, *in* 'IEEE International Conference on Robotics and Automation', Nice, France, pp. 830–835.

Matarić, M. J. (1994), Interaction and Intelligent Behavior, Technical Report AI-TR-1495, MIT Artificial Intelligence Lab.

Matarić, M. J. & Simsarian, K. T. (1995), Learning to Cooperate Using Two Six–Legged Mobile Robots, *in* 'Proceedings, Eighth European Conference on Machine Learning', Heraklion, Greece.

Noreils, F. R. (1993), 'Toward a Robot Architecture Integrating Cooperation between Mobile Robots: Application to Indoor Environment', *The International Journal of Robotics Research* **12**(1), 79–98.

Parker, L. E. (1994), Heterogeneous Multi–Robot Cooperation, PhD thesis, MIT.

Rosenschein, G. & Genesereth, M. R. (1985), Deals Among Rational Agents, *in* 'IJCAI-85', pp. 91–99.

Schwartz, J. & Sharir, M. (1983), 'On the Piano Mover's problem. II. General techniques for computing topological properties of real algebraic manifolds', *Advances in Applied Math.*