

Adaptive Multi-Robot Coordination Based on Resource Spending Velocity

(Extended Abstract)

Dan Erusalimchik¹, Gal A. Kaminka¹, Shai Shlomai², Dov Miron¹, and Sarit Kraus²
The ¹MAVERICK Group and the ²Multi-Agent Systems Group
Computer Science Department
Bar Ilan University, Israel

1. INTRODUCTION

Multi-robot systems researchers have been investigating coordination methods for improving spatial coordination in teams [2]. Such methods attempt to resolve spatial conflicts between team-members, e.g., by dynamic setting of right-of-way priorities [3], or territorial separation [1]. It is accepted that no one method is always best [2], and that all methods reach a point where adding robots to the group (i.e., increasing the density of the robots in space) reduces overall productivity [1].

There is thus growing interest in adaptive coordination approaches, which adapt the coordination methods to the dynamic changes in density. This is done, for example, by adjusting the right-away priorities based on the amount of local effort by team-members [3], or using an algorithm-selection approach [2]. In general, all of these adaptive coordination methods have demonstrated much success in multiple domains of interest.

Unfortunately, while their empirical success is evident, none of these methods have ever been analytically proven to work, nor understood in the context of existing formal work on multi-robot learning and adaptation. As a result, their optimality and the appropriate conditions for their use remain open questions. Put simply, they pose a puzzle: These are methods that work well in practice—both in simulations and with real robots—but the reasons for their success remain elusive.

This extended abstracts briefly surveys a reinforcement-learning approach to coordination algorithm selection, which not only works well in experiments, but also explored analytically. The reward function used as the basis for the learning is called *Effectiveness Index* (EI). The key idea in EI is to reduce time and resources spent coordinating, and maximize the time between conflicts that require coordination. It does this by measuring the *resource-spending velocity* (the resource "burn rate"). The use of reinforcement learning minimizes this velocity. EI does not require any knowledge of the task involved, and is thus domain-independent.

2. RELATED WORK

Most closely related is earlier work on adaptation based on coordination effort. Rosenfeld et al. [2] presented an coordination algorithm-selection approach, that adapts the selection of coordination methods by multi-robot teams, to the dynamic settings in which team-members find themselves. The method relies on mea-

suring the resources expended on coordination, using a measure called Combined Coordination Cost (CCC); however, it ignores the gains accumulated from long periods of no coordination needs, in contrast to our work.

Zuluaga and Vaughan [3] presented the *rational aggression* method for distributed robot teams. When robots come too close to each other, each of the robots picks a level of *aggression* proportional to the robot's task investment; the robot with the lowest level concedes its place. They have shown that this can produce better overall system performance compared to aggression chosen at random. This result is compatible with our findings. However, Effectiveness Index relies solely on task-independent resource measures.

3. LIMITING RESOURCE SPENDING

Coordination prevents and resolves conflicts among robots in a multi-robot system (MRS). Such conflicts can emerge as results for shared resource (e.g., space), or as a result of violation of joint decisions by team-members. However, deciding on a coordination method for use is not a trivial task, as the effectiveness of coordination methods in a given context is not known in advance.

We focus here on loosely-coupled application scenarios where coordination is triggered by conflict situations, identified through some mechanism (we assume that such a mechanism exists, though it may differ between domains; most researchers simply use a pending collision as a trigger). Thus the normal routine of a robot's operation is to carry out its primary task, until it is interrupted by an occurring or potentially-occurring conflict with another robot, which must be resolved by a coordination algorithm. Each such interruption is called a *conflict event*. The event triggers a coordination algorithm to handle the conflict. Once it successfully finishes, the robots involved go back to their primary task. Such multi-robot scenarios include foraging, search and exploration, and deliveries.

Let $A = \{\dots, a_i, \dots\}$, $1 \leq i \leq N$ be a group of N robots, cooperating on a group task that started at time 0 (arbitrarily) lasts up-to time T (A starts working and stops working on the task together). We denote by $T_i = \{c_{i,j}\}$, $0 \leq j \leq K_i$ the set of conflict events for robot i , where $c_{i,j}$ marks the time of the beginning of each conflict.

The time between the beginning of a conflict event j , and up until the next event, the interval $I_{i,j} = [c_{i,j}, c_{i,j+1})$, can be broken into two conceptual periods: The *active* interval $I_{i,j}^a = [c_{i,j}, t_{i,j})$ (for some $c_{i,j} < t_{i,j} < c_{i,j+1}$) in which the robot was actively investing resources in coordination, and the *passive* interval $I_{i,j}^p = [t_{i,j}, c_{i,j+1})$ in which the robot no longer requires investing in coordination; from its perspective the conflict event has been successfully handled, and it is back to carrying out its task. By definition $I_{i,j} = I_{i,j}^a + I_{i,j}^p$. We define the *total active time* as $I^a = \sum_i \sum_j I_{i,j}^a$ and the *total passive time* as $I^p = \sum_i \sum_j I_{i,j}^p$.

Cite as: Adaptive Multi-Robot Coordination Based on Resource Spending Velocity, (Extended Abstract), Dan Erusalimchik, Gal A. Kaminka, Shai Shlomai, Dov Miron, Sarit Kraus, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 1269–1270

Copyright © 2009, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org), All rights reserved.

Our research focuses on a case where the robot has a nonempty set M of coordination algorithms to select from. The choice of a specific coordination method $\alpha \in M$ for a given conflict event $c_{i,j}$ may effect the active and passive intervals $I_{i,j}^a, I_{i,j}^p$ (and possibly, other conflicts; see next section). To denote this dependency we use $I_{i,j}^a(\alpha), I_{i,j}^p(\alpha)$ as active and passive intervals (respectively), due to using coordination method α . Figure 1 illustrates this notation.

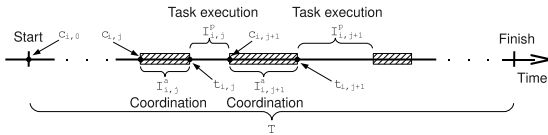


Figure 1: Illustration of task time-line, from the robots' perspective. Task execution is occasionally interrupted by the requirement to spend resources on coordination.

We define the problem of coordination algorithm selection in terms of reinforcement learning. We assume each robot tries to maximize its own reward by selecting a coordination method α . Typically, reward functions are given, and indeed most previous work focuses on learning algorithms that use the reward functions as efficiently as possible. Instead, we assume a very basic learning algorithm (a simple Q-Learning variant), and instead focus on defining a reward function (see below).

We call the proposed general reward for coordination *Effectiveness Index* (EI). Its domain independence is based on its using three intrinsic (rather than extrinsic) factors in its computation; these factors depend only on internal computation or measurement, rather than environment responses:

1. The cost of coordinating. The first factor we consider is the cost of internal resources (other than time) used by the chosen method. This is especially important in physical robots, where battery life and power are a concern.

We denote by C_i^C the total cost of coordination, of robot i . It can be broken into the costs spent on resolving all conflicts $C_i^C = \sum_j C_{i,j}^C$. $C_{i,j}^C$ is similar to other measures suggested previously, but excludes the cost of time and resources spent before the conflict (unlike [2]), and is limited to only considering individual intrinsic resources (unlike [3]).

Let us use a cost function $cost_i(\alpha, t)$ to represent the costs due to using coordination method $\alpha \in M$ at any time t during the lifetime of the robot. The function is not necessarily known to us a-priori (and indeed, in this research, is not).

Using the function $cost_i(\alpha, t)$ we define the $C_{i,j}^C$ of a particular event of robot i at time $c_{i,j}$:

$$\begin{aligned} C_{i,j}^C(\alpha) &= \int_{c_{i,j}}^{t_{i,j}} cost_i(\alpha, t) dt + \int_{t_{i,j}}^{c_{i,j+1}} cost_i(\alpha, t) dt \\ &= \int_{c_{i,j}}^{t_{i,j}} cost_i(\alpha, t) dt \end{aligned} \quad (1)$$

$C_{i,j}^C$ is defined as the cost of applying the coordination algorithm during the active interval $[c_{i,j}, t_{i,j}]$ and the passive interval $[t_{i,j}, c_{i,j+1})$. However, the coordination costs during the passive interval are zero by definition.

2. The time spent coordinating. The main goal of a coordination algorithm is to reach a (joint) decision that allows all involved robots to continue their primary activity. Therefore, the sooner the robot returns to its main task, the less time is spent on coordination, and likely, the robot can finish its task more quickly. Thus, smaller I_i^a is better. Note that this is true regardless of the use of other resources (which are measured by C_i^C). Even if somehow other resources were free, effective coordination would minimize

conflict-resolution time.

We thus define the *Active Coordination Cost* (ACC) function for robot i and method α at time $c_{i,j}$, that considers the *active time* in the calculation of coordination resources cost:

$$ACC_{i,j}(\alpha) \equiv I_{i,j}^a(\alpha) + C_{i,j}^C(\alpha) \quad (2)$$

3. The frequency of coordinating. If there are frequent interruptions to the robot's task in order to coordinate, even if short-lived and inexpensive, this would delay the robot. We assume (and the results show) that good coordination decisions lead to long durations of non-interrupted work by the robot. Therefore, the frequency of coordination method's use is not less important than the time spent on conflict resolving. Thus, larger $I_{i,j}^p$ is better.

We thus want to balance the total active coordination cost $ACC_i = \sum_j ACC_{i,j}$ against the frequency of coordination. We want to balance short-lived, infrequent calls to an expensive coordination method against somewhat more frequent calls to a cheaper coordination method.

We therefore define the Effectiveness Index of robot i , of conflict j , due to using coordination method $\alpha \in M$ as follows:

$$EI_{i,j}(\alpha) \equiv \frac{ACC_{i,j}(\alpha)}{I_{i,j}^a(\alpha) + I_{i,j}^p(\alpha)} = \frac{I_{i,j}^a(\alpha) + C_{i,j}^C(\alpha)}{I_{i,j}^a(\alpha) + I_{i,j}^p(\alpha)} \quad (3)$$

That is, the effectiveness index (EI) of a coordination method α during this event is the velocity by which it spends resources during its execution, amortized by how long a period in which no conflict occurs. Since greater EI signifies greater costs, we typically put a negation sign in front of the EI, to signify that greater velocity is worse; we seek to minimize resource spending velocity. To do this, we utilize EI as a reward function in a Q-learning algorithm.

4. AN OVERVIEW OF RESULTS

Effectiveness Index (EI) estimates the resource spending velocity of a robot, due to its efforts spent on coordination. By minimizing EI, robots dedicate more time to the task, and are thus capable of improving their team utility. We used EI as a reward function for selecting between coordination methods, by reinforcement-learning. Preliminary results demonstrate that the use of EI in Q-learning can work well in three different domains: Simulation-based multi-robot foraging, real AIBO multi-robot foraging, and high-fidelity commercial virtual environment. The experiments explore the scope of the technique, its successes and limitations. In addition, we have begun to formally explore multi-robot tasks for which EI is intended. We believe that under some assumptions, EI may emerge analytically from a game-theoretic look at the coordination in these tasks. We believe that this work represents a step towards bridging the gap between theoretical investigations of interactions, and their use to inform real-world multi-robot designs. **Acknowledgements.** This research was supported in part by ISF grant #1357/07. As always, many thanks to K. Ushi.

5. REFERENCES

- [1] M. Fontan and M. Mataric. Territorial multi-robot task division. *IEEE Transactions of Robotics and Automation*, 14(5):815–822, 1998.
- [2] A. Rosenfeld, G. A. Kaminka, S. Kraus, and O. Shehory. A study of mechanisms for improving robotic group performance. *AIJ*, 172(6–7):633–655, 2008.
- [3] M. Zuluaga and R. Vaughan. Reducing spatial interference in robot teams by local-investment aggression. In *IROS*, Edmonton, Alberta, August 2005.