Capabilities in Heterogeneous Multi-Robot Systems

Jennifer Buehler
ARC Centre of Excellence in Autonomous Systems
School of Computer Science and Engineering, UNSW
Syndey, NSW, 2052, Australia
jenniferb@cse.unsw.edu.au

Introduction

Robotic systems are desirable in many applications, particularly when the aim is to reduce human presence, for example in dangerous scenarios such as search and rescue missions. Such domains are typically dynamic and highly unpredictable, actions of robots are nondeterministic and limited bandwidth and communication failures further add to the hostility of the environment. Accurate predictions of the outcomes of a robot's actions are virtually impossible in such domains. For such robotic systems to operate in real-time, **approximative** models and algorithms are required which help to estimate the outcome with *highest possible confidence*.

Heterogeneous multi-robot systems are characterized by the diversity of the robots, each contributing different capabilities. Increasing variety of robotic systems create the need for flexible architectures enabling easy integration of new robot configurations into existing multi-robot systems. This requires methods for general reasoning about what different robots are capable of doing.

Although many aspects of heterogeneous multi-robot systems have been widely studied, few researchers explicitly formalize robot *capabilities*. A model of capabilities can prove very useful for describing and reasoning about the diversity in a multi-robot system. In this paper I present a framework that formalizes a robot's capabilities, abstracting from underlying robot architectures and providing a means to estimate a robot's performance.

Robot task suitabilities

One key element in multi-robot systems is to assign tasks to robots such that a meaningful division of work is achieved. For estimating "expected quality of task execution", *utility* is a widely used concept in multi-robot coordination. The goal of *task allocation* is to find robot-task assignments such that the overall utility is maximized (Gerkey and Matarić 2004). Many approaches to compute such a utility measure have been proposed. Only a few explicitly consider different notions of *robot capability*, relevant examples being COBOS (Fua and Ge 2005) in which each task is specified by a set of depending subtasks for which the robots learn their suitability; ASyMTRe (Parker and Tang 2006) is a framework based on *schema theory* for generating task solutions in which robots share their capabilities, considering probability of success and schema cost activation; (He and Ioerger 2003) introduce *numerical vectors of merit* describing the robot's sensing/acting, processing and communication properties which are matched to task requirements to determine a robot's suitability; Similarly, (Chen and Sun 2010) propose an algorithm in which robots and tasks are characterized by *resources* and *resource functional elements*.

Previous research has taken into account a robot's **intrinsic** capabilities[1] for estimating such utility values (Fua and Ge 2005). Simple **extrinsic** factors have also been considered, such as metric distance to the task (Stentz and Zlot 2006), or resource requirements (Chen and Sun 2010). However, such considerations were mainly tailored to the specific experiments. It has been noted that *"since different tasks demand different kinds of calculations, it is difficult to estimate factors like task-completion time to allow meaningful comparison of metrics computed by different robots, especially if task information is uncertain"* (Fua and Ge 2005). I argue that even though such estimates may be only approximate, they would significantly reduce the complexity and increase the robustness of task allocation algorithms in most realistic applications where robots contribute a variety of capabilities and are spread out over a dynamic and partially observable domain. I propose a framework which aids *estimating* task execution qualities considering task-specific details and the capabilities of robots.

Outline of my approach

I propose a model of capabilities which can provide estimates of task solution qualities at three levels of confidence. On the first level, capability requirements for a task rule out robots which are not eligible (because they don't possess the capability). On the second level, a *rough estimate* is given on how well, and if, a robot may meet the task-specific requirements. Third, more accurate estimates are given based on the result of more elaborate planning algorithms. Whereas the first level only returns a Boolean (can do or not), the second

[1] *Intrinsic* capabilities express what a robot can do in general (e.g. lift a rock), whereas *extrinsic* factors specify task details (i.e. the size/weight of the rock)

and third level will yield an estimated task-completion time and probability of success. Both time and probability values can be assigned a *variance* to further refine expressiveness.

A compact model describing capabilities *on the first two levels* can be communicated across robots, so that each robot potentially can determine teammates' eligibility for a task locally without the need to know specific details about the other robots' hardware and employed algorithms. This is highly beneficial when communication bandwidth is low or communication failures are likely. Better knowledge about teammates can reduce communication demands, or even trigger the search for a teammate currently out of range which possesses the required skills. The *third level* involving more elaborate planning mostly will have to be evaluated on-board by the robot in question and can be consulted to obtain better estimates. There is no guarantee that all robots provide the second or third level - in such cases, the lowest confidence level available provides the best estimate that can be obtained. When low-priority or non-critical tasks arise, lower probabilities of success can be traded off against possible shorter task completion time.

Details of my approach

I define a *capability* as a simple functional element which can be part of many different tasks. This is similar to how *subtasks* are defined in (Fua and Ge 2005), for which each robot learns their suitability. My definition is supported by (Zuech and Miller 1989) p. 163:

> ”There are a limited number of task types and a limited number of task decompositions [...]. There are only a few different types of **reach, grasp, lift, transport, position, insert, twist, push, pull, release**, etc. A list of *parameters* with each macro can specify *where* to reach, *when* to grasp, *how far* to twist, *how hard* to push, etc.”

A *capability* abstracts from underlying architectures at a medium level of granularity. For example, it is not important *how* a robot grasps an object (e.g. which finger movements), but only *what* it can *probably* grasp.

A listing of such required capabilities can be the result of *task decomposition*[2] algorithms. Such capabilities can have dependencies of hardware devices, and between each other. For example, *grasping* an object will always depend on *seeing* it, *localizing* it, and *reaching* for it. If any of the required capabilities is not functioning or reduced (e.g. if the image quality is reduced under current conditions), it can be inferred that the quality of depending capabilities will be reduced accordingly (i.e. grasping an object will work with lower probability).

Similar to the work by (Parker and Tang 2006) I will base my work on *schema theory* to represent the parameters and dependencies between capabilities. Each *task specification* will include a set of capabilities it depends on, similar as in COBOS (Fua and Ge 2005) which specifies for each task a set of depending subtasks. My work will distinguish from this in the way I incorporate *estimates* concerning extrinsic factors and by the *capability dependencies*.

I will now further elaborate on the three levels described in the previous section. On the **first level**, robots which do not have the required capabilities for a task are ruled out. On the **second level**, task-specific details have to be considered in an *approximate* way. For this, each capability has to be assigned *parameters* that each robot can learn[3]. For example, areas that a robot can *reach* can be approximated with a cylindrical shape around the robot; terrain a robot can *move* on may be described by indices of ”terrain roughness” with assigned average speeds; sizes it can *grasp* may be approximated with a spherical shape; *lifting* will be assigned weight ranges, and so on. While such parameters by no means allow for accurate predictions, they still provide a much better estimate than simply assuming the robot has or does not have the capability. The **third level** yields the *highest confidence predictions* because it uses more elaborate planning taking into account the algorithms employed by the robot. For example, employing path planning algorithms to better estimate the distance to the target or working out kinematics required to reach an object.

Contribution and future work

A robot's task suitability depends not only on the presence of hardware and software components, but also on how well it suits task-specific details. Dynamic and partly observable domains require approximative measures for such suitabilities. The proposed framework provides estimates on task execution qualities taking task specific details into account. Future work includes a proof-of-concept implementation, elaborating details on the computation of utility, and evaluation in a series of experiments in simulation and on real robots, comparing to traditional market-based task allocation.

References

Chen, J., and Sun, D. 2010. An online coalition based approach to solving resource constrained multirobot task allocation problem. In *2010 IEEE International Conference on Robotics and Biomimetics*, 92–97.

Fua, C., and Ge, S. 2005. COBOS: cooperative backoff adaptive scheme for multirobot task allocation. *Robotics, IEEE Transactions on* 21(6):1168–1178.

Gerkey, B. P., and Matarić, M. J. 2004. A formal analysis and taxonomy of task allocation in multi-robot systems. *The Intl. J. of Robotics Research* 23(9):939954.

He, L., and Ioerger, T. R. 2003. A quantitative model of capabilities in Multi-Agent systems. *Proceedings of IC-AI'2003* 730–736.

Parker, L., and Tang, F. 2006. Building multirobot coalitions through automated task solution synthesis. *Proceedings of the IEEE* 94:1289–1305.

Stentz, A., and Zlot, R. 2006. Market-based multirobot coordination for complex tasks. *The International Journal of Robotics Research* 25(1):73–101.

Zuech, N., and Miller, R. K. 1989. *Machine Vision*. Springer.

[2] This research is going to assume such task decomposition/planning algorithms are available. For the experiments, task descriptions are to be manually provided.

[3] Application of learning is going to be left for future work