

Meeting notes – Topic: market-based task allocation

For MAS/MRS Reading Group Meeting on 10/29/04

- Readings:
 - Stentz, Dias, Zlot, Kalra, “Market-based Approaches for Coordination of Multi-robot Teams at Different Granularities of Interaction”, American Nuclear Society 10th International Conference on Robotics and Remote Systems for Hazardous Environments, 2004
 - Additonal notes on task allocation
- When is task allocation appropriate in a multirobot system?
- Is there a difference between task and role allocation?
- How does this work differ from the task allocation paper we covered last week (Gerkey 03)?
- How do constraints on the tasks fit in to the common market-based allocation frameworks?
- Utility / cost / reward functions – How dependent is the system on the accuracy of task valuation? What about utility/cost dependencies between robots?
- What are the implications of using the “single-task-per-robot” (OAP) formulation of the problem?
- Single-item auctions vs combinatorial auctions
 - Single-task auctions: simpler, but can hit local minima pretty easily
 - Combinatorial auctions: valuation problem is hard (heuristic clusterings often used); clearing is hard; high communication complexity; but can arrive at optimal solutions (depending on the starting conditions) or avoid some local minima that single-task auctions hit
 - Other types of auctions / contracts / negotiation mechanisms
- Centralized vs peer-to-peer allocation mechanisms
 - Peer-to-peer mechanisms allows reallocation which is nice when there is uncertainty, failures, or bad initial allocations
 - Centralized mechanisms can make the system vulnerable to failures of the centralized agent or the communications system
- Can a market-based approach (as they are currently defined) be used effectively for “persistent”-type tasks?
- “Continuous vs discrete” allocation
- What can’t current market-based approaches currently/ever do?
- Other features of market-based: handling online tasks, recovering from robot failures
- How can we handle breaking contracts / failing tasks?
- Complex task allocation – how can you decompose a task before you know to whom it will be allocated, and how can you allocate a task before you know how best to decompose it?
- Learning and adaptation (*e.g.* for cost/reward calculations)
- Tight coordination: next week