Opportunistic Optimization for Market-Based Multirobot Control

M. Bernardine Dias and Anthony Stentz

Presented by: Wenjin Zhou

Why Multiple Robots?

- Some tasks require a team
  - Robotic soccer
- Some tasks can be decomposed and divided for efficiency
- Increase robustness with redundancy
- High impact on automation

The Challenge

- Enable robots to work together in an intelligent manner to execute a global task

Basic Approaches

- Centralized
- Distributed
- Market-based

Centralized Approach

- A single robot or computer is the “leader”
- Plans optimal actions for group
- Cons:
  - Computationally hard
  - response sluggish or inaccurate

Distributed Approach

- Each robot operates independently based on local sensor information
- Con:
  - solutions are often highly sub-optimal

Market Based Approach:
The Basic Idea

- Based on the economic model of a free market
- Each robot seeks to maximize individual "profit"
- Robots can negotiate and bid for tasks
- Individual profit helps the common good
- Decisions are made locally but effects approach optimality
  - Preserves advantages of distributed approach

Analogy To Real Economy

- Robots must be self-interested
- Sometimes robots cooperate, sometimes they compete
- Individuals gain benefits of their good decisions, suffer consequences of bad ones
- Just like a real market economy, the result is global efficiency

The Market Mechanism In Detail:
Background

- Consider:
  - A team of robots assembled to perform a particular set of tasks
  - Each robot is a self-interested agent
  - The team of robots is an economy
  - The goal is to complete the tasks while minimizing overall costs

How Do We Determine Profit?

- Profit = Revenue – Cost
- Team revenue is sum of individual revenues, and team cost is sum of individual costs
- Costs and revenues set up per application
  - Maximizing individual profits must move team towards globally optimal solution
- Robots that produce well at low cost receive a larger share of the overall profit

Prices and Bidding

- Robots can receive revenue from other robots in exchange for goods or services
- If robots can produce more profit together than apart, they should deal with each other
  - If one is good at finding objects and another is good at transporting them, they can both gain

How Are Prices Determined?

- Bidding
  - Robots negotiate until price is mutually beneficial
  - Note: this moves global solution towards optimum
- Robots can negotiate several deals at once
- Deals can potentially be multi-party
- Prices determined by supply and demand
  - Example: If there are a lot of movers, they won't be able to command a high price
  - This helps distribute robots among "occupations"

Competition vs. Coordination

- Complementary robots will cooperate
  - A grasper and a transporter could offer a combined “pick up and place” service
- Similar robots will compete
  - This drives prices down
- This isn’t always true:
  - Subgroups of robots could compete
  - Similar robots could agree to segment the market
  - Several grasping robots might coordinate to move a heavy objects

Contributions

- Improve market-based approach
  - Opportunistic optimization with leaders
  - Clustering for Multi-Task Processing

Optimizing with Leaders

- A robot can offer its services as a leader
- A leader investigates plans for other robots
- If it finds a way for other robots to coordinate to maximize profit:
  - Uses this profit to bid for the services of the robots
  - Keeps some profit for itself
- Allows the approach to slide along the continuum of centralized and distributed approaches in the direction of improved profitability

Clustering for Multi-Task Processing

- If robots bid on every possible combination of tasks, the number of bids submitted will grow exponentially with the number of tasks
  - Necessary to determine the clusters of tasks to bid on
- Algorithm is chosen to ensure a span in size and task membership
- Refer to the paper for details of algorithm

Why Is This Good?

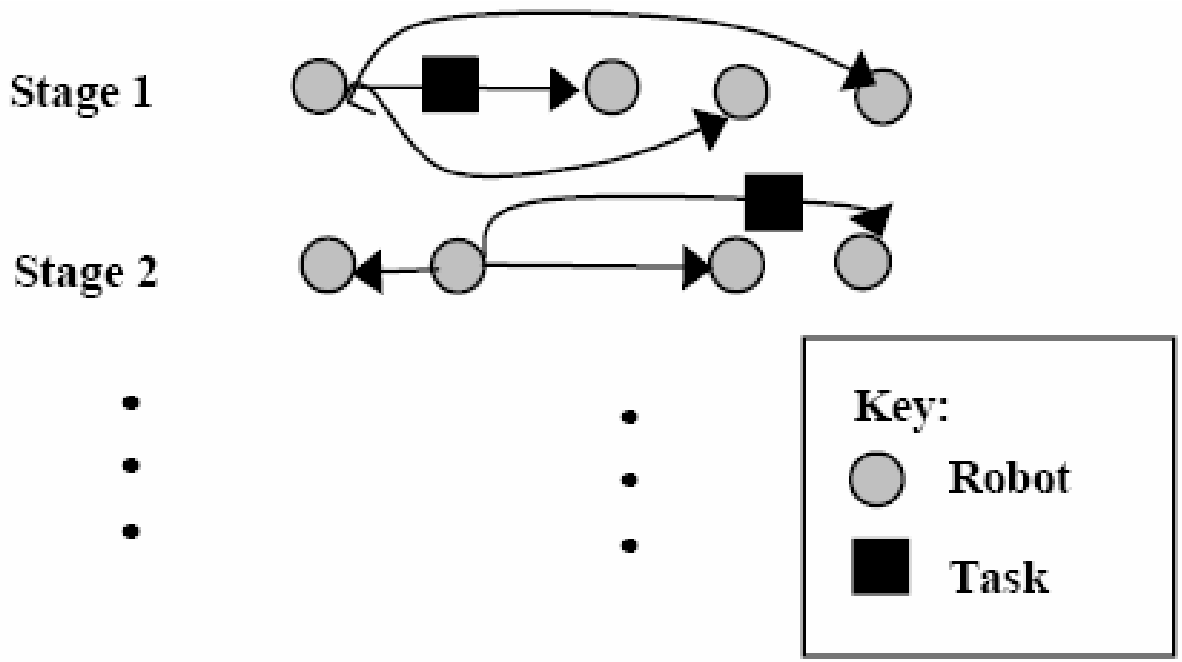
- Robust to changing conditions
  - Not hierarchical
  - If a robot breaks, tasks can be re-bid to others
- Distributed nature allows for quick response
- Only local communication necessary
- Efficient resource utilization and role adoption
- Advantages of distributed system with optimality approaching centralized system

Experimentation

- A group of robots located at different starting positions, are assigned the task of visiting a set of pre-selected observation points.
- Cases:
  - Two-party, Single-task (TPST)
  - Two-party, Multi-Task (TPMT)
  - Leader Performing Multi-party Single-task (MPST)
  - Leader Performing Multi-Party, Multi-Task (MPMT)

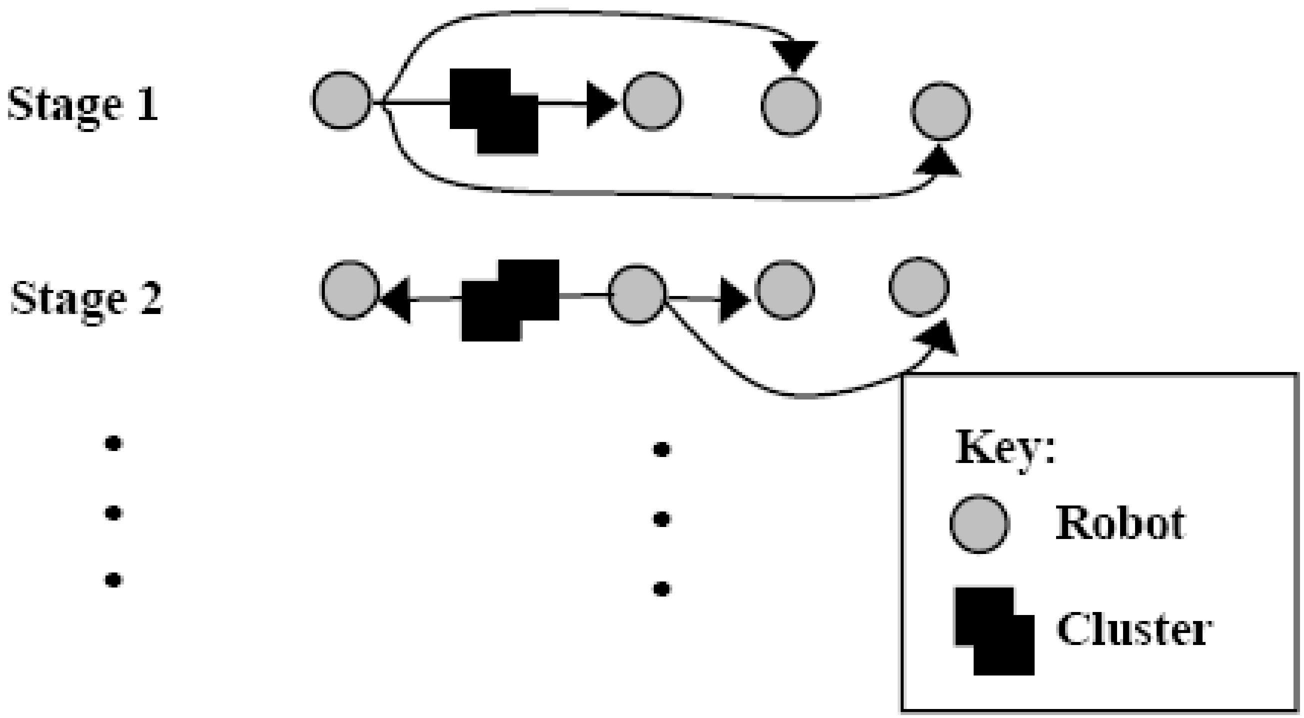
Two-party, Single-task (TPST) Negotiations

- Once the initial random task assignments are made, each of the robots, in turn, offers all its assigned tasks to all the other robots, in turn.
- Interactions are limited to **two** parties at any given time



Two-party, Multi-Task (TPMT) Negotiations

- Previous case repeated with clusters of tasks being the atomic unit of negotiations



Leader Performing Multi-party Single-task (MPST) Optimizations

- Single-task leader is introduced

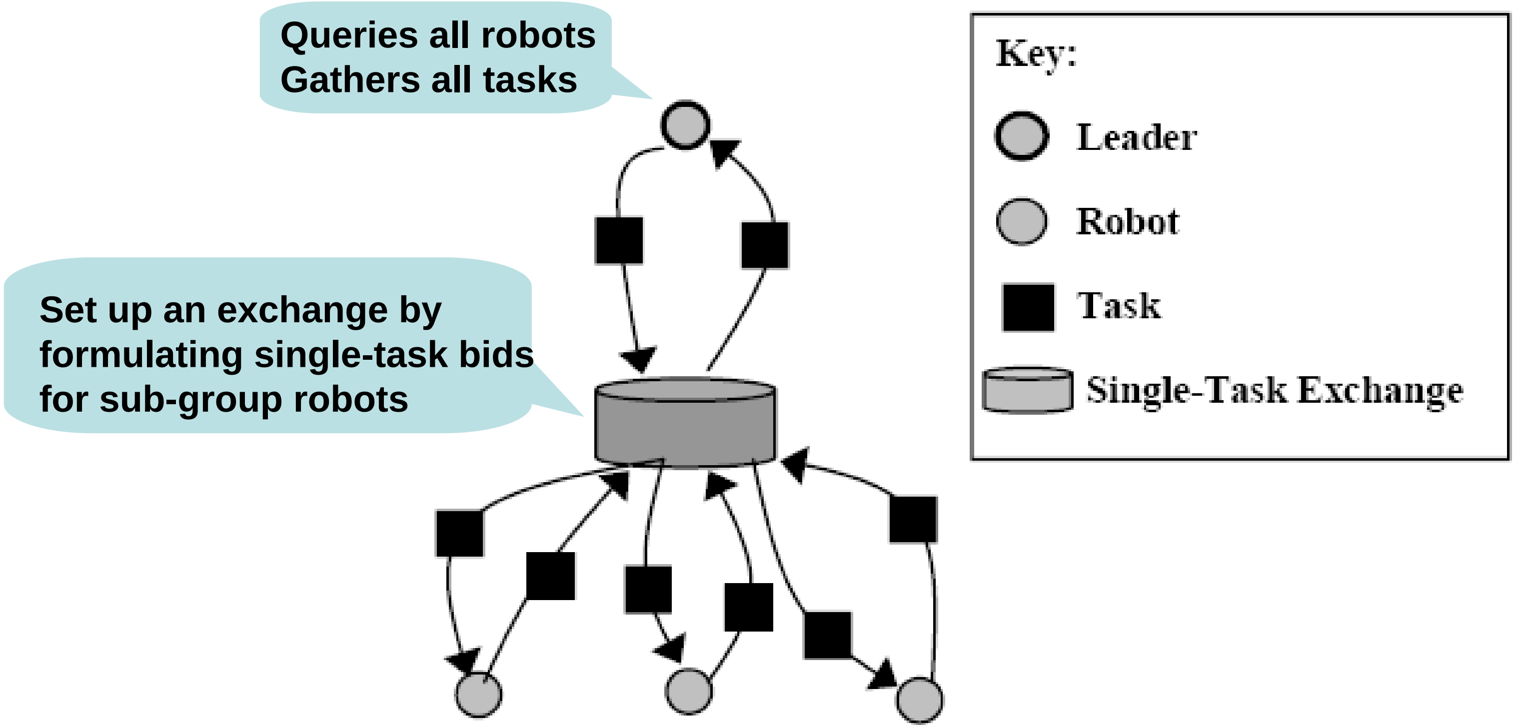
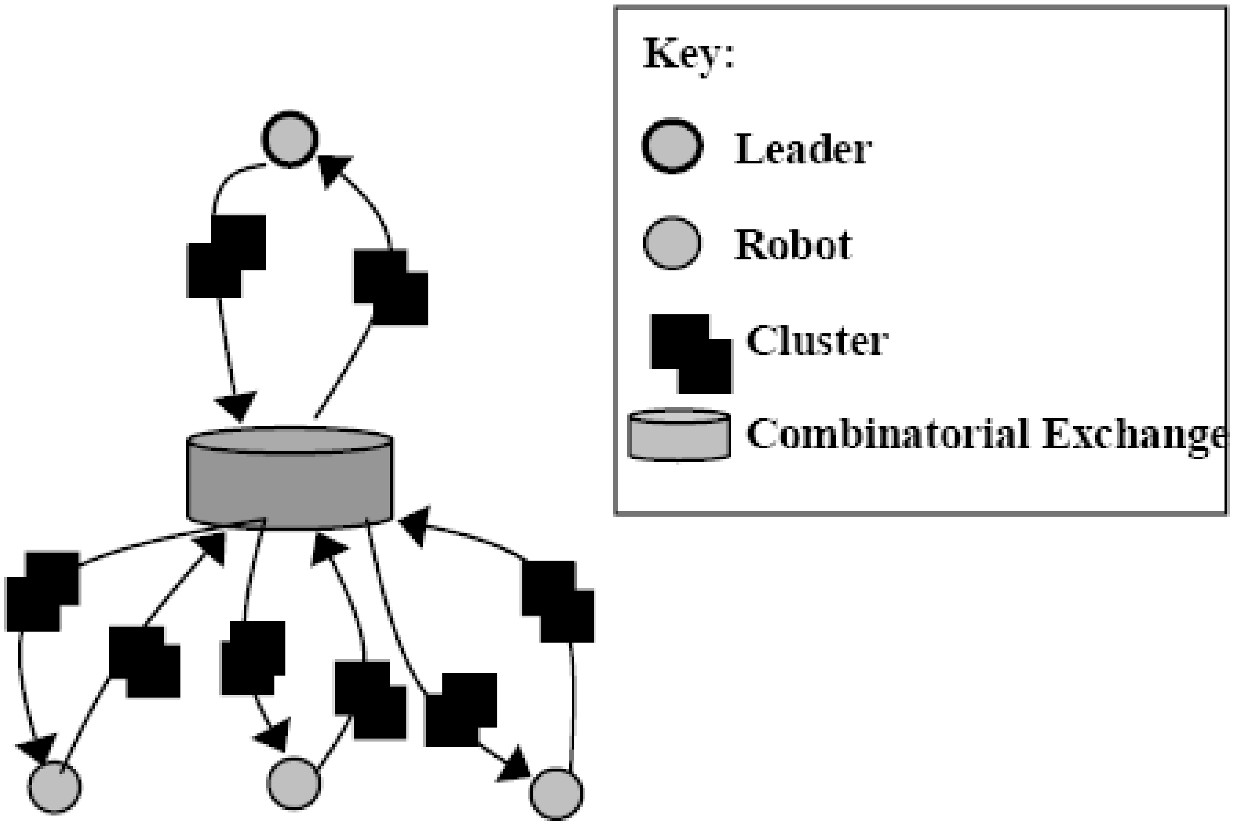


Figure 3: MPST Illustration

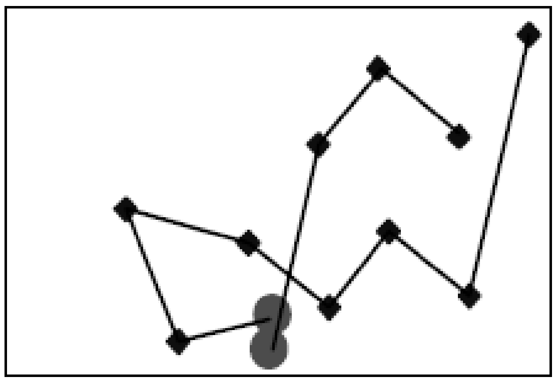
Leader Performing Multi-Party, Multi-Task (MPMT) Optimizations

- Multi-task leader is introduced

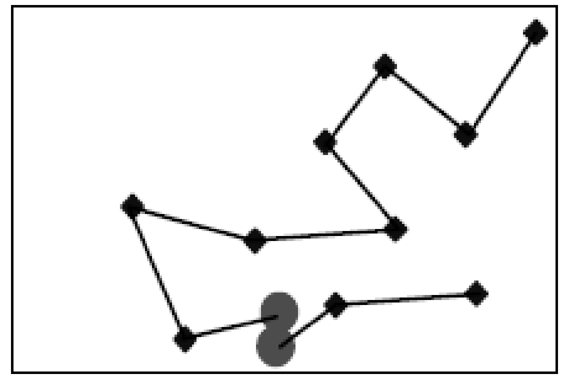


2-robot, 10-task with and without leader-optimization

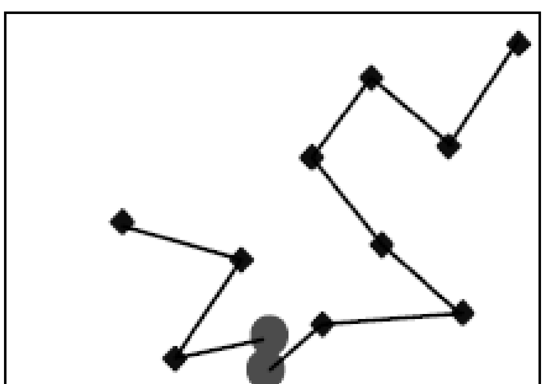
Random



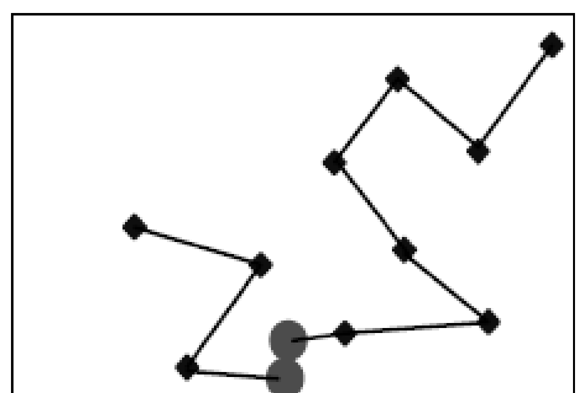
Two-Party Single-Task



Single-Task Leader



Multi-Task Leader/Optimal

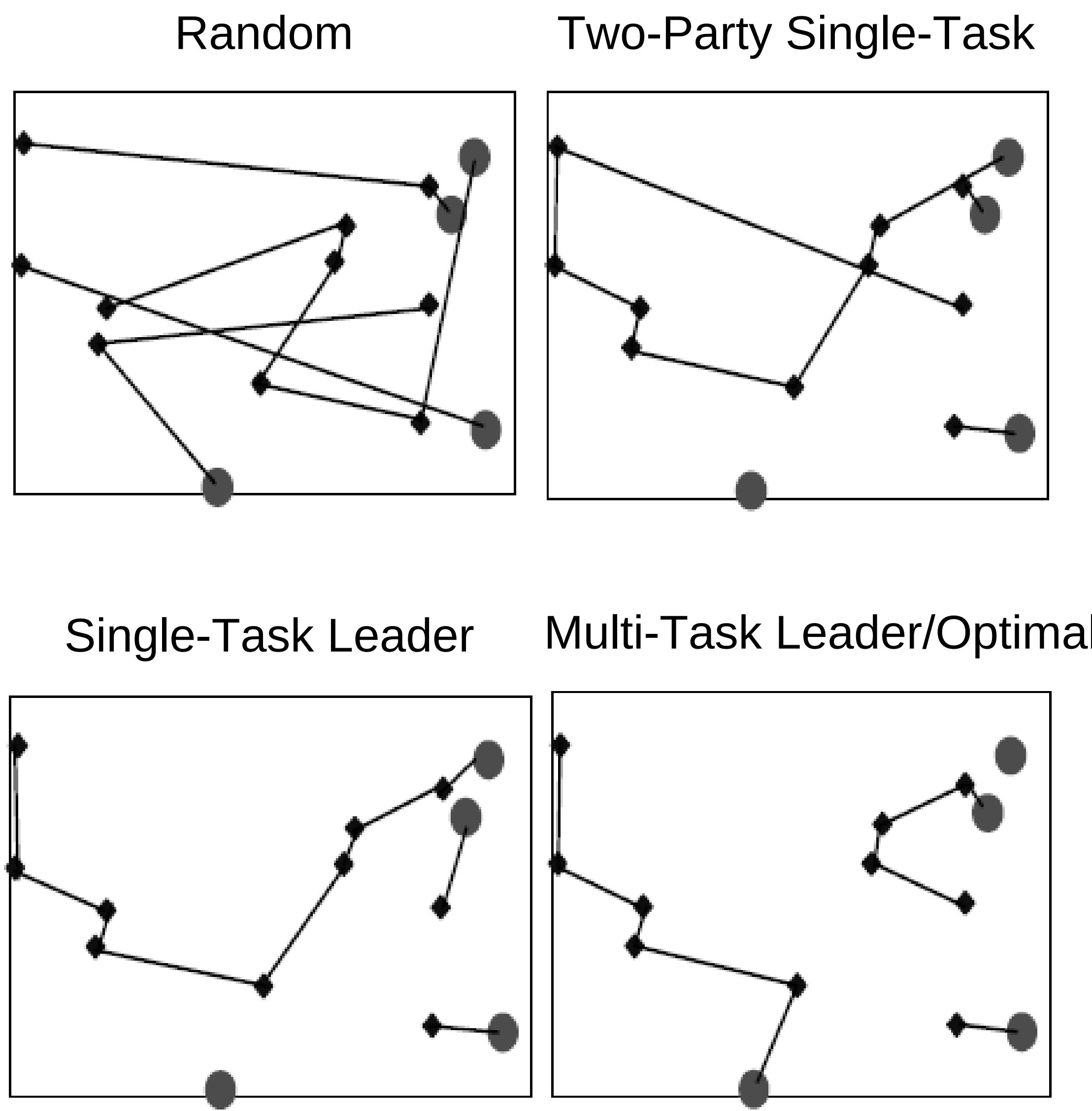


- Higher Improvement
- Lower Error

	Cost	Itns	Improved	Opt. Error
Random	351	-	0.0 %	65.6 %
No Leader				
2 ST	256	2	25.9 %	21.4 %
2 MT	231	1	33.0 %	9.0 %
ST Leader	245	2	29.0 %	16.2 %
MT Leader	227	1	34.4 %	7.0 %
Optimal	212	-	38.6 %	0.0 %

Table 1: *Performance averaged over 100 randomly generated 2-robot, 10-task TSPs*

4-robot 10-task with and without leader-optimization

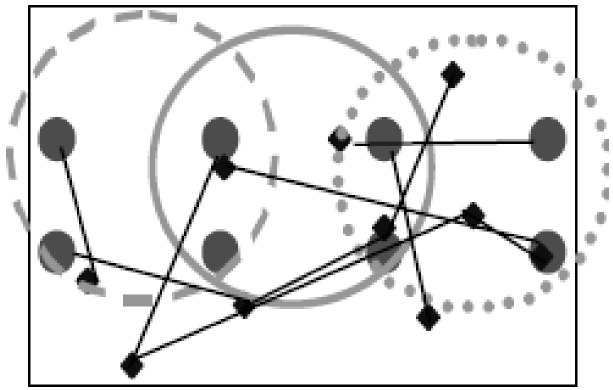


	Cost	Itns	Improved	Opt. Error
Random	411	-	0.0 %	124.6 %
No Leader				
4 ST	230	5	42.7 %	27.7 %
2ST+2MT	222	5	44.6 %	23.3 %
1ST+3MT	209	4	47.8 %	16.2 %
4MT	197	4	50.9 %	9.7 %
ST Leader	218	3	45.8 %	21.1 %
MT Leader	193	2	51.8 %	7.5 %
Optimal	183	-	-	0.0 %

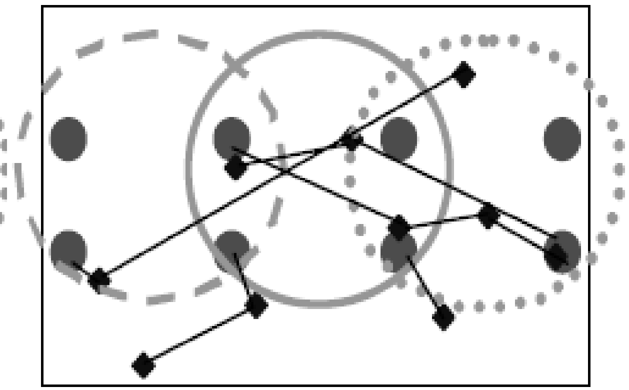
Table 2: *Results averaged over 100 randomly generated 4-robot (heterogeneous), 10-task TSPs*

3 Overlapping subgroups of 4 robots each and 10 tasks

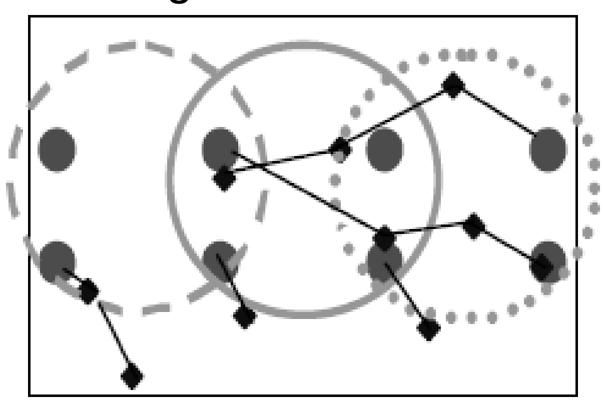
Random



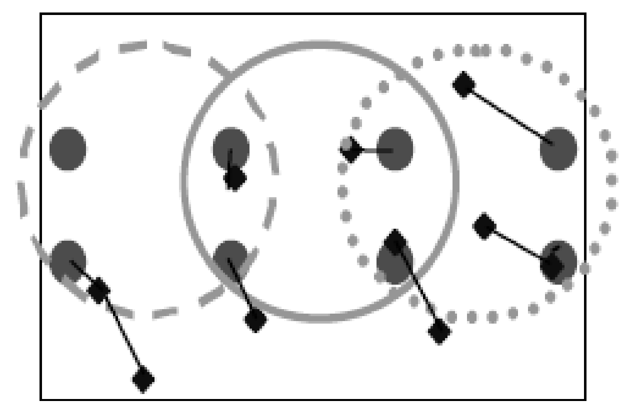
Two-Party Single-Task



Single-Task Leader



Multi-Task Leader/Optimal



	Cost	Iterations	Improved
Random	9091	-	0.0%
No Leader			
4 ST	4598	8	48.9%
2ST+2MT	4379	9	51.2%
ST Leader	4312	6	52.1%
MT Leader	3687	6	58.9%

Table 4: *Performance averaged over 100 randomly generated 8-robot (heterogeneous), 10-task TSPs with 3 overlapping groups of 4 robots each*

Thank you!