Multi-robot Coalition Formation in Real Time Scenarios

Jose Guerrero, Gabriel Oliver
Universitat de les Illes Balears, Mathematics and Computer Science Department, Palma, Spain

Presented By: Ayan Dutta
CMANTIC Lab, March, 2013.
Problem Studied

- Multi-robot Task Allocation (MRTA)
- Coalition formation.
- Auction Algorithms.
MRTA

- Why multi-robot?
- Number of tasks vs. Number of robots
- Tasks need to be completed before deadline.
- Physical interference hamper task-efficiency.
- Avoiding interference using auction.
Coalition of robots

- Coalition: A team of robots working together.
- Why do we need coalition?
- Expected coalition size determination.
- Expected time to complete the task.
- Maximization of utility.
Objective

- New MRTA method.
- Physical interference is taken into account.
- Estimating task's execution time.
- Coalition size determination.
- Improving the utility by using Duble-Round (DR) auction mechanism.
Problem Statement

- $m$ tasks and $n$ robots.
- Amount of work needed to finish task $j$: $\text{taskWorkLoad}_j$.
- Deadline: $DL_j > 0$.
- Utility function $U_j(ft_j) > 0$, where $ft_j$ is the time taken.
- Two types of deadlines: Soft and Hard.
Deadlines

(a) Hard deadline utility function.

(b) Soft deadline utility function.
Problem Statement (contd.)

- For each robot $r_i$, task $t_j$ combination; associated work capacity: $workCapacity_{i,j}$.
- Time to complete task depends on $workCapacity_{i,j}$ and physical interference.
- Coalition $g$ of robots has work capacity: $groupCapacity_{g,j}$.
- **Objective**: Maximize the utility of sum of all groups.
Complexity of the problem

- Equivalent to Set Partition Problem.
- NP-Hard problem.
- Problem can not be approximated within $O(m^{(1-e)})$, for all $e>0$.
- Thus, solution presented is also NP-Hard.
Execution Time Prediction

\[ DL_{g,j} = \frac{taskWorkLoad_j}{groupCapacity_{g,j}}. \]

\[ groupCapacity_{g,j} = idealCapacity_{g,j} - 1 \]

\[ idealCapacity_{g,j} = \sum_{1 \leq i \leq n_g} workCapacity_{i,j} \]

work capacity \( \left( \frac{taskWorkLoad_j}{T_{i,j}} \right) \) is:

\[ workCapacity_{i,j} = \frac{loadCapacity_i \cdot v_i}{2 \cdot (loadCapacity_i v_i + d_j)}. \]
Predicting Interference

Off-line Method:
1. *Support Vector Regression* (SVR) method is used.
2. Better than classical regression.
3. Regression: Relationship between variables; specially among dependent and independent variables.

\[
f(x) = \sum_{1 \leq i \leq n_v} (\alpha_i - \alpha_i^*) K(x, x_i) + b
\]

\[
K(x, x_i) = e^{-\gamma \|x-x_i\|^2}
\]
**Algorithm 1** Leadership request algorithm. Robot $r_1$ requests the leadership of task $t$

**Require:** $t$: task

1: Send a request for $t$
2: $time_{init} \leftarrow time$
3: while $time - time_{init} \leq TIME_{LEADERSHIP}$ do
4: \hspace{1em} if new message from $r_2$ then
5: \hspace{2em} if $r_2$ is already the $t$ leader then
6: \hspace{3em} Give up the request process
7: \hspace{3em} return
8: \hspace{2em} end if
9: \hspace{2em} if $r_2$ requests $t$ and $r_2$ is better than $r_1$ then
10: \hspace{3em} Give up the request process
11: \hspace{3em} return
12: \hspace{2em} end if
13: \hspace{2em} if $r_2$ requests $t$ and $r_1$ is better than $r_2$ then
14: \hspace{3em} Continue
15: \hspace{2em} end if
16: end if
17: end while
18: $r_1$ is the new task $t$ leader
Auction for a Task

Algorithm 2 Leader’s auction algorithm.

Require: $t_j$: task to execute
Ensure: List $R_p$ of selected robots and the AWARD message.
1: Send a bid request for the task $t_j$
2: $B_r \leftarrow \emptyset$
3: $time_{init} \leftarrow time$
4: while $time - time_{init} \leq TIME \text{ AUCTION}$ do
5: \hspace{1em} if a new bit $b_i$ received from robot $i$ then
6: \hspace{2em} $B_r \leftarrow B_r \cup b_i$
7: \hspace{1em} end if
8: end while
9: Sort $B_r$ from higher to lower work capacity
10: $DL_{g,j} \leftarrow \infty$
11: $k \leftarrow 0$
12: $R_p \leftarrow \emptyset$
13: while $i \leq |B_r|$ and $DL_{g,j} \geq DL_j$ do
14: \hspace{1em} $R_p \leftarrow R_p \cup (\text{robot's bid } B_r[k])$
15: \hspace{1em} Update $DL_{g,j}$ using the equation 6
16: \hspace{1em} $i \leftarrow i + 1$
17: end while
18: for all $r \in R_p$ do
19: \hspace{1em} Bid for the robot $r$ with the value $U_j(DL_{g,j})$
20: end for
No Leader's Auction

Algorithm 3  No leader robot's auction for a robot algorithm

1: if AWARD message, $b_0$, from a leader then
2: \[ B_l \leftarrow b_0 \]
3: \[ time_{init} \leftarrow time \]
4: while \( time - time_{init} \leq \) TIME_BID_ACCEPTED do
5: \hspace{1em} if AWARD message, $b_i$, from a leader then
6: \hspace{2em} \[ B_l \leftarrow B_l \cup b_i \]
7: \hspace{1em} end if
8: end while
9: \[ l_{best} \leftarrow \text{best leader in } B_l \]
10: Send a ROBOT_ALIVE message to $l_{best}$
11: for all $l_i$ in $B_l$ and $l_i \neq l_{best}$ do
12: \hspace{1em} Send REFUSE message to $l_i$
13: end for
14: end if
Complexity Comparison

Computational and communication complexity of several MRTA algorithms.

<table>
<thead>
<tr>
<th>Method</th>
<th>Computational</th>
<th>Communication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Double Round auction</td>
<td>$A : O(\max(n \log(n), n^3))$</td>
<td>$A : O(m)$</td>
</tr>
<tr>
<td></td>
<td>$B : O(n)$</td>
<td>$B : O(m)$</td>
</tr>
<tr>
<td>[33] Service et al.'10</td>
<td>$O(n^{2j} \cdot m)$</td>
<td>-</td>
</tr>
<tr>
<td>[23] Vig et al.'06</td>
<td>$O(n^k m)$</td>
<td>-</td>
</tr>
<tr>
<td>Dynamic Role [25]</td>
<td>$A : O(n)$</td>
<td>$A : O(n)$</td>
</tr>
<tr>
<td></td>
<td>$B : O(1)$</td>
<td>$B : O(1)$</td>
</tr>
</tbody>
</table>
Experimental Settings

Summary of the experiments carried out.

<table>
<thead>
<tr>
<th></th>
<th>M_I</th>
<th>NM_I</th>
<th>M_NI</th>
<th>NM_NI</th>
</tr>
</thead>
<tbody>
<tr>
<td>DR</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>SR</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Greedy</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

- **M_I**: the continual monitoring strategy is used together with the interference model.
- **NM_I**: the interference model is used, but without monitoring the progress of the task.
- **M_NI**: the continual monitoring strategy is used but without the interference method. That is, the expected execution time is calculated using only the ideal capacity.
- **NM_NI**: neither continual monitoring nor interference model are used.
Simulation Setting
Results

(a) Greedy results.  
(b) NM_NI results.
Contd.

(c) NM_I results.

(d) M_NI results.

(e) M_I results.
Percentage of tasks executed before the deadline (column $t \leq DL$) and tasks that need less than 10% of its deadline to finish. The deadline is equal to 900 units and the utility function is homogeneous.

<table>
<thead>
<tr>
<th></th>
<th>$t \leq DL$</th>
<th>$t &lt; (DL + 10%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greedy</td>
<td>30.87</td>
<td>42.57</td>
</tr>
<tr>
<td>NM_NI</td>
<td>12.56</td>
<td>23.86</td>
</tr>
<tr>
<td>NM_I</td>
<td>36.67</td>
<td>54.27</td>
</tr>
<tr>
<td>M_NI</td>
<td>16.16</td>
<td>37.36</td>
</tr>
<tr>
<td>M_I</td>
<td>15.18</td>
<td>30.38</td>
</tr>
</tbody>
</table>
Heterogeneous Tasks

Percentage of tasks executed before the deadline (column $t \leq DL$) and tasks that need less than 10% of its deadline to finish (column $t \leq (DL + 10\%)$). The deadline is equal to 900 units and the utility function is heterogeneous following the Eq. (10). The double round auction process has been used in all cases.

<table>
<thead>
<tr>
<th></th>
<th>$t \leq DL$</th>
<th>$t &lt; (DL + 10%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greedy</td>
<td>37.88</td>
<td>48.48</td>
</tr>
<tr>
<td>NM_NI</td>
<td>25.30</td>
<td>37.40</td>
</tr>
<tr>
<td>NM_I</td>
<td>29.30</td>
<td>50.50</td>
</tr>
<tr>
<td>M_NI</td>
<td>25.25</td>
<td>55.05</td>
</tr>
<tr>
<td>M_I</td>
<td>26.76</td>
<td>50.00</td>
</tr>
</tbody>
</table>
Varying Deadline

**Fig. 8.** Total utility with the soft deadline function.
Real Robot Platform

(a) Pioneer-3DX.
(b) The four robots used during the experiments.
Conclusions

- New auction method presented.
- DR auction works better than previous algorithms.
- Estimating coalition size.
- Predicting task finish time.
- Real time restriction consideration: interference.