Organizational Self-Design in Semi-Dynamic Environments

Sachin Kamboj and Keith S. Decker

Department of Computer and Information Sciences
University of Delaware
Newark, DE 19716

Sixth International Joint Conference on Autonomous Agents and Multiagent Systems - (AAMAS 2007)
1. Introduction
   - Motivation
   - Problem Representation

2. Approach
   - Agent roles and relationships
   - Organization Formation and Adaptation

3. Evaluation
   - Types of Experiments
   - Comparison with the Contract Net Protocol
   - Evaluation of the three task allocation heuristics
Outline

1. Introduction
   - Motivation
   - Problem Representation

2. Approach
   - Agent roles and relationships
   - Organization Formation and Adaptation

3. Evaluation
   - Types of Experiments
   - Comparison with the Contract Net Protocol
   - Evaluation of the three task allocation heuristics
Introduction
Basic Model

Problem Instance
Instance # 1
Deadline: 40
Task:

Method 5
Method 4
Method 3
Method 2
Method 1
Task D
Task B
Task C
Task A
Method 6

MULTIAGENT SYSTEM
SOLUTIONS
Introduction

Problems

Problem 1: Build Car 1
Instance of Problem 2: Build Car 2
Instance of Problem 3: Build Car 3
Instance of Problem 4: Build Car 4

Build a Car

Build the parts
- Build the engine
- Frame the body
- Build the transmission

Assemble the Car
Environmental conditions
- Problems arrive at varying rates
- Problems might have different deadlines
- Available resources might change over time
Environmental conditions

- Problems arrive at varying rates
- Problems might have different deadlines
- Available resources might change over time
Environmental conditions

- Problems arrive at varying rates
- Problems might have different deadlines
- Available resources might change over time
Introduction

Basic Model

Problem Instance
Instance # 1
Deadline: 40
Task:

Method 5
Method 4
Method 3
Method 2
Task D
Method 1
Task B Task C
Task A
Method 6

TIME

MULTIAGENT SYSTEM

SOLUTIONS
Organizational Issues

- How many agents do we need?
- How do we allocate subtasks and resources to the agents?
- How do we coordinate inter-agent activities?
Organizational Issues

- How many agents do we need?
- How do we allocate subtasks and resources to the agents?
- How do we coordinate inter-agent activities?
Organizational Issues

- How many agents do we need?
- How do we allocate subtasks and resources to the agents?
- How do we coordinate inter-agent activities?
So what’s the problem?

- There is no best way of organizing and all ways of organizing are not equally effective
- Environmental may be dynamic
- All problem instances and environmental conditions are not unique
Introduction

Motivation

So what's the problem?

- There is no best way of organizing and all ways of organizing are not equally effective

- Environmental may be dynamic

- All problem instances and environmental conditions are not unique
So what’s the problem?

- There is no best way of organizing and all ways of organizing are not equally effective
- Environmental may be dynamic
- All problem instances and environmental conditions are not unique
Our Answer:

Problem Instance
Instance # 1
Deadline: 40
Task:

TIME

MULTIAGENT SYSTEM

SOLUTIONS
Our Answer:
We propose to use a dynamic run-time approach to organization based on Organizational-Self Design (OSD) [Ishida, Gasser, et al.]

- OSD means agents design their own organizational structures
- Our main contribution is extending existing OSD research to incorporate TÆMS based task structures.
Outline

1 Introduction
   - Motivation
   - Problem Representation

2 Approach
   - Agent roles and relationships
   - Organization Formation and Adaptation

3 Evaluation
   - Types of Experiments
   - Comparison with the Contract Net Protocol
   - Evaluation of the three task allocation heuristics
Problems are represented using TÆMS
Problems are represented using TÆMS
Introduction

Problem Representation

Task A

Task B

Task C

Task D

Sachin Kamboj, Keith S. Decker

OSD
TÆMS is quantitative

Method 1
Outcome: 1
Probability: 1.0
Quality: 6
Cost: 7
Duration: 32

Method 2
Outcome: 1
Probability: 1.0
Quality: 10
Cost: 5
Duration: 17

Method 3
Outcome: 1
Probability: 1.0
Quality: 12
Cost: 7
Duration: 12

Method 4
Outcome: 1
Probability: 0.5
Quality: 3
Cost: 4
Duration: 2
Outcome: 2
Probability: 0.5
Quality: 12
Cost: 10
Duration: 3

Method 5
Outcome: 1
Probability: 0.7
Quality: 10
Cost: 5
Duration: 20
Outcome: 2
Probability: 0.3
Quality: 3
Cost: 7
Duration: 30

Method 6
Outcome: 1
Probability: 1.0
Quality: 10
Cost: 5
Duration: 20
Outcome: 2
Probability: 0.3
Quality: 3
Cost: 7
Duration: 30
TÆMS is quantitative

Method 1
Outcome: 1
Probability: 1.0
Quality: 6
Cost: 7
Duration: 32

Method 2
Outcome: 1
Probability: 1.0
Quality: 10
Cost: 5
Duration: 17

Method 3
Outcome: 1
Probability: 1.0
Quality: 12
Cost: 7
Duration: 12

Method 4
Outcome: 1
Probability: 0.5
Quality: 3
Cost: 4
Duration: 2
Outcome: 2
Probability: 0.5
Quality: 12
Cost: 10
Duration: 3

Method 5
Outcome: 1
Probability: 0.7
Quality: 10
Cost: 5
Duration: 20
Outcome: 2
Probability: 0.3
Quality: 3
Cost: 7
Duration: 30

Method 6
Outcome: 1
Probability: 1.0
Quality: 12
Cost: 7
Duration: 7
Outcome: 2
Probability: 0.5
Quality: 0.5
Cost: 4
Duration: 2
Outcome: 3
Probability: 0.5
Quality: 12
Cost: 10
Duration: 3

Sachin Kamboj, Keith S. Decker
OSD
Introduction

Problem Representation

Method 1
- Outcome: 1
- Probability: 1.0
- Quality: 6
- Cost: 7
- Duration: 32

Method 2
- Outcome: 1
- Probability: 1.0
- Quality: 10
- Cost: 5
- Duration: 17
- Probability: 1.0
- Quality: 10
- Cost: 5
- Duration: 17

Method 3
- Outcome: 1
- Probability: 1.0
- Quality: 12
- Cost: 7
- Duration: 12

Method 4
- Outcome: 1
- Probability: 0.5
- Quality: 3
- Cost: 4
- Duration: 2
- Outcome: 2
- Probability: 0.5
- Quality: 12
- Cost: 10
- Duration: 3

Method 5
- Outcome: 1
- Probability: 0.7
- Quality: 10
- Cost: 5
- Duration: 20
- Outcome: 2
- Probability: 0.3
- Quality: 3
- Cost: 7
- Duration: 30

Method 6
- Outcome: 1
- Probability: 1.0
- Quality: 12
- Cost: 7
- Duration: 7

Task A
- MAX

Task B
- SUM

Task C
- SEQ_SUM

Task D
- MIN

Facilitates: MIN

Enables: MAX

Sachin Kamboj, Keith S. Decker
OSD
Outline

1. Introduction
   - Motivation
   - Problem Representation

2. Approach
   - Agent roles and relationships
   - Organization Formation and Adaptation

3. Evaluation
   - Types of Experiments
   - Comparison with the Contract Net Protocol
   - Evaluation of the three task allocation heuristics
Organizational design is directly contingent on:

1. The task structure
2. The environmental conditions under which the problems need to be solved
Agent roles and relationships

Organizational structure is primarily composed of:
- Roles:
  - Parts played by the agents enacting the roles in the solution to the problem
- Relationships:
  - Coordination relationships that exist between the subparts of a problem

Definition
A role is a TÆMS subtree rooted at a particular node
Organizational structure is primarily composed of:

- **Roles:** Parts played by the agents enacting the roles in the solution to the problem

- **Relationships:** Coordination relationships that exist between the subparts of a problem

**Definition**

A role is a TÆMS subtree rooted at a particular node.
Organizational structure is primarily composed of

- **Roles:**
  - Parts played by the agents enacting the roles in the solution to the problem

- **Relationships:**
  - Coordination relationships that exist between the subparts of a problem

**Definition**

A role is a TÆMS subtree rooted at a particular node
Agent roles and relationships

Organizational structure is primarily composed of

- **Roles:**
  - Parts played by the agents enacting the roles in the solution to the problem
- **Relationships:**
  - Coordination relationships that exist between the subparts of a problem

**Definition**

A role is a TÆMS subtree rooted at a particular node
Agent roles and relationships

Task A

Task B

Task C

Task D

Method 1

Method 2

Method 3

Method 4

Method 5

Method 6

Enables

MAX

SUM

SEQ_SUM

Facilitates

MIN
Agent roles and relationships

Task A

Task B

Task C

Task D

Method 1

Method 2

Method 3

Method 4

Method 5

Method 6

Role 1

Role 2

MAX

SUM

SEQ_SUM

Enables

Facilitates

MIN

Sachin Kamboj, Keith S. Decker

OSD
Agent roles and relationships

Task A
- MAX

Task B
- Enables
- SUM

Task C
- SEQ_SUM

Task D
- MIN

Role 1
- Facilitates

Role 2

Role 3
- Method 3

Role 4
- Method 4
- Method 5
- Method 6

Method 1
- Facilitates
- MIN

Method 2

Method 3

Method 4

Method 5

Method 6
Outline

1. Introduction
   - Motivation
   - Problem Representation

2. Approach
   - Agent roles and relationships
   - Organization Formation and Adaptation

3. Evaluation
   - Types of Experiments
   - Comparison with the Contract Net Protocol
   - Evaluation of the three task allocation heuristics
Organization Formation and Adaptation

- Start off with an initial organization consisting of a single agent, solely responsible for all activities
- Each agent in the organization checks to see if:
  - It is overloaded:
    - It spawns off a new agent to handle part of its load
  - It is free (underloaded)
    - It combines with another agent to save resources
Start off with an initial organization consisting of a single agent, solely responsible for all activities.

Each agent in the organization checks to see if:

- It is overloaded:
  - It spawns off a new agent to handle part of its load.
- It is free (underloaded):
  - It combines with another agent to save resources.
Start off with an initial organization consisting of a single agent, solely responsible for all activities.

Each agent in the organization checks to see if:
- It is overloaded:
  - It spawns off a new agent to handle part of its load
- It is free (underloaded)
  - It combines with another agent to save resources
Start off with an initial organization consisting of a single agent, solely responsible for all activities.

Each agent in the organization checks to see if:

- It is overloaded:
  - It spawns off a new agent to handle part of its load
- It is free (underloaded):
  - It combines with another agent to save resources
Organization Formation and Adaptation

- Start off with an initial organization consisting of a single agent, solely responsible for all activities
- Each agent in the organization checks to see if:
  - It is overloaded:
    - It spawns off a new agent to handle part of its load
  - It is free (underloaded)
    - It combines with another agent to save resources
Start off with an initial organization consisting of a single agent, solely responsible for all activities.

Each agent in the organization checks to see if:

- It is overloaded:
  - It spawns off a new agent to handle part of its load
- It is free (underloaded)
  - It combines with another agent to save resources
Agent Spawning

AGENT 1

Task A

Task B

Task C

Task D

Method 1

Method 2

Method 3

Method 4

Method 5

Method 6

Organization 1
Agent Spawning

**Organization Formation and Adaptation**

**Agent roles and relationships**

Introduction
Approach
Evaluation

**Agent Spawning**

**MIN**

**Method 1**

**Task A**

**MAX**

**Method 2**

**SUM**

**Method 3**

**SEQ_SUM**

**Method 4**

**Method 5**

**Method 6**

**Agent Spawning**

**Organization 1**

**Cost (Task B) = 10**

Sachin Kamboj, Keith S. Decker

OSD
Organization Formation and Adaptation

Agent Spawning

Agent roles and relationships

Organization Formation and Adaptation

Introduction
Approach
Evaluation

Agent Spawning

Task A

Task B

Task C

Task D

Method 1

Method 2

Method 3

Method 4

Method 5

Method 6

AGENT 1

MAX

SUM

Enables

SEQ_SUM

Facilitates

MIN

Cost (Task C) = 4

Sachin Kamboj, Keith S. Decker

OSD
Agent Spawning

Cost (Method 1) = 7
Organization Formation and Adaptation

Agent Spawning

Agent roles and relationships
Organization Formation and Adaptation

Introduction
Approach
Evaluation

Task A
Task B
Task C
Task D

Method 1
Method 2
Method 3
Method 4
Method 5
Method 6

MAX
SUM
SEQ_SUM

AGENT 1

Cost (Task D) = 8

Sachin Kamboj, Keith S. Decker
OSD
Agent Spawning

- Task A
  - MAX
  - Enables
    - Task B
      - SUM
      - Enables
        - Task C
          - SEQ_SUM
          - Method 2
          - Method 3
    - Method 1
    - Task D
      - MIN
      - Facilitates
        - Method 4
        - Method 5
    - Method 6

Lowest Cost = Cost(Task C)
Agent Spawning

Agent 1

- Task A
  - Task B
    - Method 1
  - Task D
    - Method 4

Agent 2

- Task C
  - Method 2
  - Method 3

Enables

- MAX
- SUM
- SEQ_SUM

Facilitates

- MIN

Organization 2
What is the cost function?

- We evaluated three cost functions (based on three heuristics):
  1. Allocating top-most roles first
  2. Minimizing total resource cost
  3. Balancing execution time
The three cost functions

AGENT 1

- Task A
  - Task B
    - Method 1: Duration: 3, Resource A: 5
  - Task C
    - Method 2: Duration: 10, Resource A: 9
    - Method 3: Duration: 2, Resource A: 7

SUM

MIN

MIN
Organization Formation and Adaptation

Agent Spawning

Topmost First

- Number the roles while doing a BFS.
- Cost(Role) = Number assigned to Role
**Organization Formation and Adaptation**

**Agent Spawning**

**Topmost First**
- Number the roles while doing a BFS.
- $\text{Cost}(\text{Role}) = \text{Number assigned to Role}$

**Diagram**

- **Task A**
  - **Task B**
    - **Method 1**
      - Duration: 3
      - Resource A: 5
  - **Method 2**
    - Duration: 10
    - Resource A: 9
  - **Method 3**
    - Duration: 2
    - Resource A: 7
  - **Method 4**
    - Duration: 4
    - Resource B: 5

- **Task C**

**Summary**

- **Method 1**
  - Duration: 3
  - Resource A: 5
- **Method 2**
  - Duration: 10
  - Resource A: 9
- **Method 3**
  - Duration: 2
  - Resource A: 7
- **Method 4**
  - Duration: 4
  - Resource B: 5
Topmost First

- Number the roles while doing a BFS.
- Cost(Role) = Number assigned to Role

Method 1
Duration: 3
Resource A: 5

Method 2
Duration: 10
Resource A: 9

Method 3
Duration: 2
Resource A: 7

Method 4
Duration: 4
Resource B: 5
Minimizing Resource Cost

\[ \text{Cost}(\text{Role}) = \text{Resource Cost}(\text{Role}) + \text{Resource Cost}(\text{Role}) \]

**Agent Spawning**

**Task A**
- Method 1: Duration: 3, Resource A: 5
- Method 2: Duration: 10, Resource A: 9
- Method 3: Duration: 2, Resource A: 7
- Method 4: Duration: 4, Resource B: 5

**Task B**
- **SUM**

**Task C**
- **MIN**

**AGENT 1**

Sachin Kamboj, Keith S. Decker
OSD
Minimizing Resource Cost

\[ \text{Cost}(Role) = \text{Resource Cost}(Role) + \text{Resource Cost}(Role) \]
Minimizing Resource Cost

- \( \text{Cost}(Role) = \text{Resource Cost}(Role) + \text{Resource Cost}(Role) \)

**Agent Spawning**

**Task**

- **Task A**
  - **Method 1**
    - Duration: 3
    - Resource A: 5
  - **Method 2**
    - Duration: 10
    - Resource A: 9
  - **Method 3**
    - Duration: 2
    - Resource A: 7
  - **Method 4**
    - Duration: 4
    - Resource B: 5

**SUM**

**MIN**

**ASSIGNMENT**

- **Agent 1**
  - **Task B**
    - **Task A**
      - **Task C**
        - **Task B**
          - **Task C**
            - **Method 1**
              - Duration: 3
              - Resource A: 5
            - **Method 2**
              - Duration: 10
              - Resource A: 9
            - **Method 3**
              - Duration: 2
              - Resource A: 7
            - **Method 4**
              - Duration: 4
              - Resource B: 5

**Cost(Task B) = 9 + 12 = 21**

Sachin Kamboj, Keith S. Decker  
OSD
Minimizing Resource Cost

\[ \text{Cost}(\text{Role}) = \text{Resource Cost}(\text{Role}) + \text{Resource Cost}(\text{Role}) \]

\begin{itemize}
  \item AGENT 1
\end{itemize}

\begin{itemize}
  \item Task A
  \item Task B
  \item Task C
\end{itemize}

\begin{itemize}
  \item Method 1
    \begin{itemize}
      \item Duration: 3
      \item Resource A: 5
    \end{itemize}
  \item Method 2
    \begin{itemize}
      \item Duration: 10
      \item Resource A: 9
    \end{itemize}
  \item Method 3
    \begin{itemize}
      \item Duration: 2
      \item Resource A: 7
    \end{itemize}
  \item Method 4
    \begin{itemize}
      \item Duration: 4
      \item Resource B: 5
    \end{itemize}
\end{itemize}

\[ \text{Cost}(\text{Method 4}) = 5 + 9 = 14 \]
Minimizing Resource Cost

\[
\text{Cost}(\text{Role}) = \text{Resource Cost}(\text{Role}) + \text{Resource Cost}(\text{Role})
\]

AGENT 1

Task A

Task B

Method 1
Duration: 3
Resource A: 5

Method 2
Duration: 10
Resource A: 9

Method 3
Duration: 2
Resource A: 7

SUM

Task C

Method 4
Duration: 4
Resource B: 5

MIN

MIN

AGENT 2

Sachin Kamboj, Keith S. Decker
OSD
Balancing Execution Time

Cost(Role) = | Expt Duration(Role) - Expt Duration(Role) |

Method 1
Duration: 3
Resource A: 5

Method 2
Duration: 10
Resource A: 9

Method 3
Duration: 2
Resource B: 7

Method 4
Duration: 4
Resource B: 5

SUM
MIN
MIN

Task A

Task B

Task C

AGENT 1

Task A

Sachin Kamboj, Keith S. Decker

OSD
Organization Formation and Adaptation

Agent Spawning

Balancing Execution Time

\[ \text{Cost}(\text{Role}) = | \text{Expt Duration}(\text{Role}) - \text{Expt Duration}(\overline{\text{Role}}) | \]

Method 1
- Duration: 3
- Resource A: 5

Method 2
- Duration: 10
- Resource A: 9

Method 3
- Duration: 2
- Resource A: 7

Method 4
- Duration: 4
- Resource B: 5

**AGENT 1**

**ED = 13**

**SUM MIN**

**Task A**

**Task B**
- **ED = 13**

**Task C**
- **ED = 6**
Balancing Execution Time

- \( \text{Cost}(Role) = | \text{Expt Duration}(Role) - \text{Expt Duration}(Role) | \)

\[
\begin{align*}
\text{Task A} & \quad \text{ED} = 13 \\
\text{Task B} & \quad \text{ED} = 13 \\
\text{Task C} & \quad \text{ED} = 6 \\
\end{align*}
\]

**Method 1**
- Duration: 3
- Resource A: 5

**Method 2**
- Duration: 10
- Resource A: 9

**Method 3**
- Duration: 2
- Resource A: 7

**Method 4**
- Duration: 4
- Resource B: 5

\[\text{Cost(Task B)} = 13 - 6 = 7\]
Balancing Execution Time

\[ \text{Cost}(Role) = | \text{Expt Duration}(Role) - \text{Expt Duration}(\overline{Role}) | \]
Balancing Execution Time

Cost(Role) = | Expt Duration(Role) - Expt Duration(Role) |

Agent Spawning

Method 1
Duration: 3
Resource A: 5

Method 2
Duration: 10
Resource A: 9

Method 3
Duration: 2
Resource A: 7

Method 4
Duration: 4
Resource B: 5

Agent 1

Agent 2

Sachin Kamboj, Keith S. Decker
OSD
Organization Formation and Adaptation

Agent Composition

Organization $i$

AGENTS 1
- Task A
  - MAX
- Task B
  - SUM
- Method 1

AGENTS 2
- Task C
  - SEQ_SUM
  - Facilitates MIN
- Method 2
- Method 3

AGENTS 3
- Task D
  - Facilitates MIN
- Method 4

AGENTS 4
- Method 5
- Method 6

Sachin Kamboj, Keith S. Decker
OSD
Organization Formation and Adaptation

Agent Composition

- Task A
  - MAX
  - Enables
  - Task C
    - SEQ_SUM
    - Facilitates
    - AGENT 1
    - AGENT 2
    - AGENT 4

- Task B
  - SUM

- Task D
  - MIN

- Method 1
- Method 2
- Method 3
- Method 4
- Method 5
- Method 6

Organization \( i+1 \)

Sachin Kamboj, Keith S. Decker
OSD
Introduction

Motivation
Problem Representation

Approach
Agent roles and relationships
Organization Formation and Adaptation

Evaluation
Types of Experiments
Comparison with the Contract Net Protocol
Evaluation of the three task allocation heuristics
We conducted 3 sets of experiments to evaluate our approach:

1. Compared with the Contract Net Protocol (CNP)
2. Evaluated the three task allocation heuristics
3. Tested the robustness of our approach
We conducted 3 sets of experiments to evaluate our approach:

1. Compared with the Contract Net Protocol (CNP)
2. Evaluated the three task allocation heuristics
3. Tested the robustness of our approach
We conducted 3 sets of experiments to evaluate our approach:

1. Compared with the Contract Net Protocol (CNP)
2. Evaluated the three task allocation heuristics
3. Tested the robustness of our approach
Outline

1. Introduction
   - Motivation
   - Problem Representation

2. Approach
   - Agent roles and relationships
   - Organization Formation and Adaptation

3. Evaluation
   - Types of Experiments
   - Comparison with the Contract Net Protocol
   - Evaluation of the three task allocation heuristics
Evaluation
Comparison with the Contract Net Protocol

The chart compares various performance metrics across different agents and task allocation heuristics. The metrics include:

- Agents
- Completed Tasks (10s)
- Messages Sent (1000s)
- Quality
- Response Time
- Resources (10s)

The heuristics evaluated are:

- ORG
- CN8
- CN10
- CN12
- CN14

The chart shows the mean performance and variability for each metric across the evaluated heuristics, allowing for a comparison of their effectiveness.
Outline

1 Introduction
   - Motivation
   - Problem Representation

2 Approach
   - Agent roles and relationships
   - Organization Formation and Adaptation

3 Evaluation
   - Types of Experiments
   - Comparison with the Contract Net Protocol
   - Evaluation of the three task allocation heuristics
Evaluation
Evaluation of the three task allocation heuristics

Control Variables:
- Task structure depth
- Branching factor
- Probability of the CAFs
- The arrival rate
- The deadline slack
Evaluation
Evaluation of the three task allocation heuristics

Experimental Setup
- Each experiment was repeated 20 times
  - With a new randomly generated task structure
  - These 20 experiments formed an experimental set.
- A static environment was used in each experiment
- The final evaluation was done on 673 experimental sets.
Evaluation
Evaluation of the three task allocation heuristics

Experimental Setup

- Each experiment was repeated 20 times
  - With a new randomly generated task structure
  - These 20 experiments formed an experimental set.
- A static environment was used in each experiment
- The final evaluation was done on 673 experimental sets.
Evaluation
Evaluation of the three task allocation heuristics

Experimental Setup
- Each experiment was repeated 20 times
  - With a new randomly generated task structure
  - These 20 experiments formed an experimental set.
- A static environment was used in each experiment
- The final evaluation was done on 673 experimental sets.
Evaluation
Evaluation of the three task allocation heuristics

Experimental Setup
- Each experiment was repeated 20 times
  - With a new randomly generated task structure
  - These 20 experiments formed an experimental set.
- A static environment was used in each experiment
- The final evaluation was done on 673 experimental sets.
Experimental Setup

- Each experiment was repeated 20 times
  - With a new randomly generated task structure
  - These 20 experiments formed an *experimental set*.
- A static environment was used in each experiment
- The final evaluation was done on 673 experimental sets.
Evaluation
Evaluation of the three task allocation heuristics

Tested Performance Criteria

1. The average number of agents used
2. The total number of organizational changes
3. The total messages sent by all the agents
4. The total resource cost of the organization
5. The number of tasks completed
6. The average quality accrued
7. The average response time of the organization
8. The average runtime of the tasks
9. The turnaround time
Evaluation
Evaluation of the three task allocation heuristics

- We ran the **Wilcoxon Matched-Pair Signed-Rank** tests on the experiments in each set.
- Null Hypothesis: *there is no difference between the pair of heuristics for the performance criteria under consideration*
  - Interested in the cases in which the null hypothesis can be rejected with 95% confidence ($p < 0.05$).
- We noted the number of times that a heuristic performed the best or was in a group that performed statistically better than the rest.
We ran the Wilcoxon Matched-Pair Signed-Rank tests on the experiments in each set.

Null Hypothesis: *there is no difference between the pair of heuristics for the performance criteria under consideration*

- Interested in the cases in which the null hypothesis can be rejected with 95% confidence ($p < 0.05$).

We noted the number of times that a heuristic performed the best or was in a group that performed statistically better than the rest.
We ran the Wilcoxon Matched-Pair Signed-Rank tests on the experiments in each set.

Null Hypothesis: *there is no difference between the pair of heuristics for the performance criteria under consideration*

Interested in the cases in which the null hypothesis can be rejected with 95% confidence ($p < 0.05$).

We noted the number of times that a heuristic performed the best or was in a group that performed statistically better than the rest.
Evaluation
Evaluation of the three task allocation heuristics

- We ran the Wilcoxon Matched-Pair Signed-Rank tests on the experiments in each set.
- Null Hypothesis: *there is no difference between the pair of heuristics for the performance criteria under consideration*
  - Interested in the cases in which the null hypothesis can be rejected with 95% confidence ($p < 0.05$).
- We noted the number of times that a heuristic performed the best or was in a group that performed statistically better than the rest.
Evaluation

Evaluation of the three task allocation heuristics

- Balancing Execution Time
- Topmost First
- Minimizing Resources
- Random Allocation

<table>
<thead>
<tr>
<th>No of Agents</th>
<th>No Org Changes</th>
<th>No Agent Changes</th>
<th>Total Messages Sent</th>
<th>Resource Cost</th>
<th>Tasks Completed</th>
<th>Average Response Time</th>
<th>Average Runtime</th>
<th>Average Turnaround Time</th>
<th>Average Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
</tr>
<tr>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>600</td>
<td>600</td>
<td>600</td>
<td>600</td>
<td>600</td>
<td>600</td>
<td>600</td>
<td>600</td>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td>700</td>
<td>700</td>
<td>700</td>
<td>700</td>
<td>700</td>
<td>700</td>
<td>700</td>
<td>700</td>
<td>700</td>
<td>700</td>
</tr>
</tbody>
</table>
Questions?