# Robotics Seminar on Multi Agent Path Finding (236824)

**Oren Salzman**

Computer Science Department, Technion

# Today's lecture

- What is the seminar about?
- Learning objectives
- Admin + Class structure
- Brief introduction to algorithmic approaches
  - SAT-based methods
  - Search-based methods

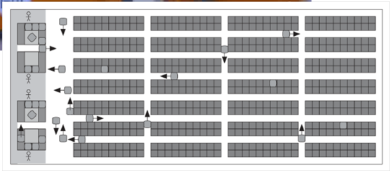
Some of the slides are adapted from slides by Sven Koenig—http://mapf.info/index.php/Main/Tutorials

# Motivation—Amazon fulfillment centers

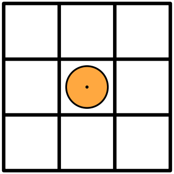
- 2003 Kiva Systems founded
- 2012 Amazon acquires Kiva Systems for $775 million
- 2015 Kiva Systems becomes Amazon Robotics


- $> 3,000$ robots on $> 110,000$ square meters in Tracy, California

# Motivation—Amazon fulfillment centers



# MAPF—problem formulation

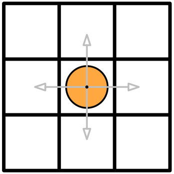



Simplifying assumptions:

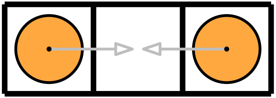
- Point robots
- No kinematic constraints
- Discretized environment

# MAPF—problem formulation (cont.)

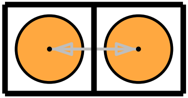
- Each agent moves N, E, S, or W into an adjacent unblocked cell



- Vertex collision



- Edge collision



# Optimization criteria

## Definition

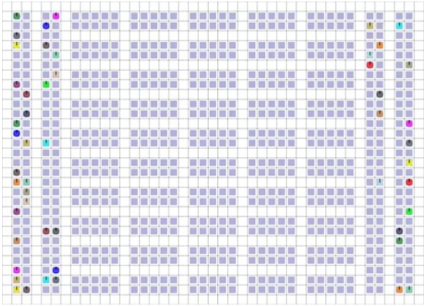
The sum of costs of a MAPF problem is the summation, over all agents, of the number of time steps required to reach the goal

## Definition

The makespan of a MAPF problem is the total time until the last agent reaches its destination (i.e., the maximum of the individual costs)

# Multi Agent Pickup & Delivery (MAPD)

- Agents have to attend to a stream of delivery tasks
- One agent has to be assigned to each delivery task
- This agent has to first move to a given pickup location and then to a given delivery location while avoiding collisions with other agents



# MAPF and MAPD in academia

- Active research topic with publications in
  - AI conferences (AAAI, IJCAI, ICAPS, SoCS, ECAI, AAMAS and more)
  - Robotic conferences (IROS, ICRA, MRS, RSS and more)
  - Leading journals (RaL, TRo, JAIR, JAAMAS, AI and more)
- Strong Israeli presence (specifically the research group from BGU)
- Vibrant community with communal resources—http://mapf.info/
  - Benchmarks
  - Publications
  - Researchers
  - Software
  - Tutorials

# MAPF and MAPD in academia

- Active research topic with publications in
  - AI conferences (AAAI, IJCAI, ICAPS, SoCS, ECAI, AAMAS and more)
  - Robotic conferences (IROS, ICRA, MRS, RSS and more)
  - Leading journals (RaL, TRo, JAIR, JAAMAS, AI and more)
- Strong Israeli presence (specifically the research group from BGU)
- Vibrant community with communal resources—http://mapf.info/
  - Benchmarks
  - Publications
  - Researchers
  - Software
  - Tutorials

# MAPF and MAPD in academia

- Active research topic with publications in
  - AI conferences (AAAI, IJCAI, ICAPS, SoCS, ECAI, AAMAS and more)
  - Robotic conferences (IROS, ICRA, MRS, RSS and more)
  - Leading journals (RaL, TRo, JAIR, JAAMAS, AI and more)
- Strong Israeli presence (specifically the research group from BGU)
- Vibrant community with communal resources—http://mapf.info/
  - Benchmarks
  - Publications
  - Researchers
  - Software
  - Tutorials

# (some) Key publications

- Peter R. Wurman, Raffaello D'Andrea, Mick Mountz: Hundreds of Cooperative, Autonomous Vehicles in Warehouses. AI Magazine 29(1):9-20 (2008)
- Guni Sharon, Roni Stern, Ariel Felner, Nathan R. Sturtevant: Conflict-based search for optimal multi-agent pathfinding. Artif. Intell. 219: 40-66 (2015)
- Hang Ma, Jiaoyang Li, T. K. Satish Kumar, Sven Koenig: Lifelong Multi-Agent Path Finding for Online Pickup and Delivery Tasks. AAMAS 2017: 837-845

# (some) Key publications

- Peter R. Wurman, Raffaello D'Andrea, Mick Mountz: Hundreds of Cooperative, Autonomous Vehicles in Warehouses. AI Magazine 29(1):9-20 (2008)
- Guni Sharon, Roni Stern, Ariel Felner, Nathan R. Sturtevant: Conflict-based search for optimal multi-agent pathfinding. Artif. Intell. 219: 40-66 (2015)
- Hang Ma, Jiaoyang Li, T. K. Satish Kumar, Sven Koenig: Lifelong Multi-Agent Path Finding for Online Pickup and Delivery Tasks. AAMAS 2017: 837-845

# (some) Key publications

- Peter R. Wurman, Raffaello D'Andrea, Mick Mountz: Hundreds of Cooperative, Autonomous Vehicles in Warehouses. AI Magazine 29(1):9-20 (2008)
- Guni Sharon, Roni Stern, Ariel Felner, Nathan R. Sturtevant: Conflict-based search for optimal multi-agent pathfinding. Artif. Intell. 219: 40-66 (2015)
- Hang Ma, Jiaoyang Li, T. K. Satish Kumar, Sven Koenig: Lifelong Multi-Agent Path Finding for Online Pickup and Delivery Tasks. AAMAS 2017: 837-845

# (some) Key publications (cont.)

- Hang Ma, Sven Koenig: AI buzzwords explained: multi-agent path finding (MAPF). AI Matters 3(3): 15-19 (2017)
- Roni Stern, Nathan R. Sturtevant, Ariel Felner, Sven Koenig, Hang Ma, Thayne T. Walker, Jiaoyang Li, Dor Atzmon, Liron Cohen, T. K. Satish Kumar, Roman Bartak, Eli Boyarski: Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks. SOCS 2019: 151-159
- Oren Salzman, Roni Stern: Research Challenges and Opportunities in Multi-Agent PathFinding and Multi-Agent Pickup and Delivery Problems, AAMAS 2020 (to appear).

# (some) Key publications (cont.)

- Hang Ma, Sven Koenig: AI buzzwords explained: multi-agent path finding (MAPF). AI Matters 3(3): 15-19 (2017)
- Roni Stern, Nathan R. Sturtevant, Ariel Felner, Sven Koenig, Hang Ma, Thayne T. Walker, Jiaoyang Li, Dor Atzmon, Liron Cohen, T. K. Satish Kumar, Roman Bartak, Eli Boyarski: Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks. SOCS 2019: 151-159
- Oren Salzman, Roni Stern: Research Challenges and Opportunities in Multi-Agent PathFinding and Multi-Agent Pickup and Delivery Problems, AAMAS 2020 (to appear).

# Learning objectives

- Exposure to state-of-the-art in MAPF and MAPD
- Academic presentation skills
- Identifying research opportunities in the field of MAPF and MAPD

# Admin & Seminar mechanics

- Office hours: Just shoot me an email.
- Requirements:
  - Mini introductory project
  - Presenting 1 scientific paper
  - Attendance in at least 80% of the oral presentations
  - Active participation
  - Writing a report at the end of the course
- Prerequisites: Intro to AI (236501) or willingness to learn A* independently

# Admin & Seminar mechanics (cont.)

Grade structure:

- 20%—CBS project
- 50%—Paper presentation
  - 17.5%—Clarity of oral presentation
  - 17.5%—Depth of presentation: insights, design considerations, connection to big picture
  - 7.5%—Response to QnA
  - 7.5%—Scholarship (references, formatting, typos, consistency in formatting etc.)
- 30%—Final report (submission by 12.7, no late submissions)

# CBS Project

- Hands-on project in Python
- Based on http://idm-lab.org/project-p/project.html

| Date | Week | Presentor | Topic / paper | Task | Deliverables |
|---|---|---|---|---|---|
| 18.3 | 1 | Oren Salzman | Introduction to MAPF | (1) Read papers [1-3]<br>(2) Review A* if needed (e.g., using [4-6])<br>(3) Implement Task (0) and (1) in [7] | Post questions / problems on [0] by 24.3 |
| 25.3 | 2 | Oren Salzman / Class | Reviewing tasks (0) and (1) | (1) Read papers [8-9]<br>(2) Post CBS-related questions on [0] by 31.3<br>(3) Implement Task (2.1-2.2) in [7] | Submit answers for tasks (0) and (1) by 29.3 |
| 1.4 | 3 | Oren Salzman / Class | Go over CBS<br>Review task (2) | (1) Implement Task (3) in [7] | Submit answers for task 2 (2.1-2.5) by 7.4 |
| 8.4 | Passover Break | | | | |
| 15.4 | Passover Break | | | | |
| 22.4 | 4 | Oren Salzman / Class | Go over CBS<br>Review task (3) | (1) Implement Tasks (4) and (5) in [7] | Submit answers for task (3) by 26.4 |
| 29.4 | Independence Day | | | | ` |
| 6.5 | 5 | Student presentation (1)<br>Student presentation (2) | | | Submit answers for task (4) and (5) by 3.5 |

# Presentation guidelines

- Presentation topic
  - Pick any paper(s) from http://mapf.info/index.php/Main/Publications
  - If you find relevant work not in that list, great!
  - Regardless, get the topic approved it in advance.
- Presentation structure
  - Roughly 30 min talk + 10 min QnA (interleaved)
  - Roughly 10 min feedback
- Fit the talk to the time, not the other way around
  - It is more than fine to go over more than one paper
  - It is even better to dive deeply into less than one paper

# Presentation guidelines

- Presentation topic
  - Pick any paper(s) from
    http://mapf.info/index.php/Main/Publications
  - If you find relevant work not in that list, great!
  - Regardless, get the topic approved it in advance.
- Presentation structure
  - Roughly 30 min talk + 10 min QnA (interleaved)
  - Roughly 10 min feedback
- Fit the talk to the time, not the other way around
  - It is more than fine to go over more than one paper
  - It is even better to dive deeply into less than one paper

# Presentation guidelines

- Presentation topic
  - Pick any paper(s) from http://mapf.info/index.php/Main/Publications
  - If you find relevant work not in that list, great!
  - Regardless, get the topic approved it in advance.
- Presentation structure
  - Roughly 30 min talk + 10 min QnA (interleaved)
  - Roughly 10 min feedback
- Fit the talk to the time, not the other way around
  - It is more than fine to go over more than one paper
  - It is even better to dive deeply into less than one paper

# Presentation tips

- Each presentation should be as self-contained as possible
- Hourglass
- Language—English is preferred (both written and oral)
- Sensory channels—2 visual (you and slides / board) + audio
- More is not always better (beware of input overload)—keep it simple
- What is the take-home message from your talk?

# Feedback

- Giving feedback—what was good? what could be improved?
  - Concrete
  - Concise
  - Constructive
- Receiving feedback
  - Listen, Listen, Listen
  - Don't be defensive
  - Thank the feedback provider

# Feedback

- Giving feedback—what was good? what could be improved?
    - Concrete
    - Concise
    - Constructive
- Receiving feedback
    - Listen, Listen, Listen
    - Don't be defensive
    - Thank the feedback provider

# Final report

- Pick a paper / topic that you did not present
- Summarize the paper in your own words
- Beyond summary—what are the key takeaways? insights? missing details? direction for future work?
- Design in detail a project around the paper
  - What are the outcomes (implement the approach is not a good project)?
  - What are the expected results?
  - What tools are required to implement the project?
  - Sanity check
    - Would you be excited to do this project (feasible, challenging, interesting)
    - Is it clear what needs to be done and how?

# Final report

- Pick a paper / topic that you did not present
- Summarize the paper in your own words
- Beyond summary—what are the key takeaways? insights? missing details? direction for future work?
- Design in detail a project around the paper
  - What are the outcomes (implement the approach is not a good project)?
  - What are the expected results?
  - What tools are required to implement the project?
  - Sanity check
    - Would you be excited to do this project (feasible, challenging, interesting)
    - Is it clear what needs to be done and how?

# Final report

- Pick a paper / topic that you did not present
- Summarize the paper in your own words
- Beyond summary—what are the key takeaways? insights? missing details? direction for future work?
- Design in detail a project around the paper
  - What are the outcomes (implement the approach is not a good project)?
  - What are the expected results?
  - What tools are required to implement the project?
  - Sanity check
    - Would you be excited to do this project (feasible, challenging, interesting)
    - Is it clear what needs to be done and how?

# Final report

- Pick a paper / topic that you did not present
- Summarize the paper in your own words
- Beyond summary—what are the key takeaways? insights? missing details? direction for future work?
- Design in detail a project around the paper
  - What are the outcomes (implement the approach is not a good project)?
  - What are the expected results?
  - What tools are required to implement the project?
  - Sanity check
    - Would you be excited to do this project (feasible, challenging, interesting)
    - Is it clear what needs to be done and how?

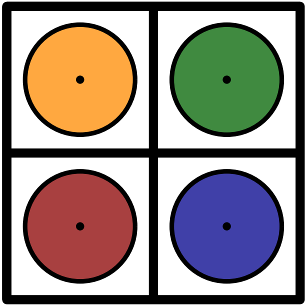
# Some hardness results

- 15-puzzle and its generalizations are NP hard [Goldreich 11, Ratner, Warmuth 90]
- Even if an agent can move to a vertex that is being left by another agent the problem is still NP hard [Surynek 10, Yu, LaValle 13]


Figure adapted from https://en.wikipedia.org/wiki/15_puzzle.

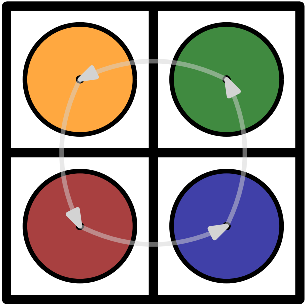
# Some hardness results

- 15-puzzle and its generalizations are NP hard [Goldreich 11, Ratner, Warmuth 90]
- Even if an agent can move to a vertex that is being left by another agent the problem is still NP hard [Surynek 10, Yu, LaValle 13]



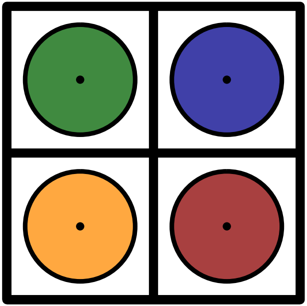
# Some hardness results

- 15-puzzle and its generalizations are NP hard [Goldreich 11, Ratner, Warmuth 90]
- Even if an agent can move to a vertex that is being left by another agent the problem is still NP hard [Surynek 10, Yu, LaValle 13]



# Some hardness results

- 15-puzzle and its generalizations are NP hard [Goldreich 11, Ratner, Warmuth 90]
- Even if an agent can move to a vertex that is being left by another agent the problem is still NP hard [Surynek 10, Yu, LaValle 13]



# SAT-based methods [Surynek 14]

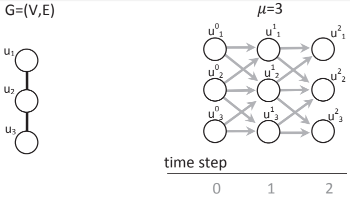
## Definition

The time expansion graph of depth $\mu$ of a graph $G = (V, E)$ is a graph $G_\mu = (V_\mu, E_\mu)$ such that

$$V_\mu = \{u_j^t \mid t \in \{0, \dots, \mu\} \text{ and } u_j \in V\}$$

and

$$E_\mu = \{(u_i^t, u_j^{t+1}) \mid t \in \{0, \dots, \mu - 1\} \text{ and } (u_i, u_j) \in E\}$$



# SAT-based methods [Surynek 14]

SAT Encoding for bounded MAPF under makespan

- Define propositional variables $x^t_{j,k}$ s.t. $x^t_{j,k}$ is assigned true if agent $k$ appears at vertex $v_j$ at time step $t$
- Define constraints modeling validity conditions
  - An agent $k$ is placed in exactly one vertex at each time step $t$
  - At most one agent $k$ is placed in each vertex $j$ at each time step $t$
  - An agent $k$ relocates to some of its neighbors or makes no move at every time step $t$
- Obtain a conjunctive normal form (CNF)
- Solve by trying for bounds $\mu = 1, 2, \ldots$ until the optimal makespan is encountered

# SAT-based methods [Surynek 14]

SAT Encoding for bounded MAPF under makespan

- Define propositional variables $x^t_{j,k}$ s.t. $x^t_{j,k}$ is assigned true if agent $k$ appears at vertex $v_j$ at time step $t$
- Define constraints modeling validity conditions
  - An agent $k$ is placed in exactly one vertex at each time step $t$
  - At most one agent $k$ is placed in each vertex $j$ at each time step $t$
  - An agent $k$ relocates to some of its neighbors or makes no move at every time step $t$
- Obtain a conjunctive normal form (CNF)
- Solve by trying for bounds $\mu = 1, 2, \ldots$ until the optimal makespan is encountered

# SAT-based methods [Surynek 14]

SAT Encoding for bounded MAPF under makespan

- Define propositional variables $x^t_{j,k}$ s.t. $x^t_{j,k}$ is assigned true if agent $k$ appears at vertex $v_j$ at time step $t$
- Define constraints modeling validity conditions
  - An agent $k$ is placed in exactly one vertex at each time step $t$

$$\bigwedge_{j \neq \ell} \left( \neg x^t_{j,k} \vee \neg x^t_{\ell,k} \right) \qquad \text{for every time } t \text{ and every agent } k$$

$$\bigvee_{j=1}^{n} x^t_{j,k} \qquad \text{for every time } t \text{ and every agent } k$$

  - At most one agent $k$ is placed in each vertex $j$ at each time step $t$
  - An agent $k$ relocates to some of its neighbors or makes no move at every time step $t$
- Obtain a conjunctive normal form (CNF)
- Solve by trying for bounds $\mu = 1, 2, \ldots$ until the optimal makespan is encountered

# SAT-based methods [Surynek 14]

SAT Encoding for bounded MAPF under makespan

- Define propositional variables $x^t_{j,k}$ s.t. $x^t_{j,k}$ is assigned true if agent $k$ appears at vertex $v_j$ at time step $t$
- Define constraints modeling validity conditions
  - An agent $k$ is placed in exactly one vertex at each time step $t$
  - At most one agent $k$ is placed in each vertex $j$ at each time step $t$

$$\bigwedge_{k \neq h} (\neg x^t_{j,k} \vee \neg x^t_{j,h}) \qquad \text{for every time } t \text{ and every vertex } j$$

  - An agent $k$ relocates to some of its neighbors or makes no move at every time step $t$
- Obtain a conjunctive normal form (CNF)
- Solve by trying for bounds $\mu = 1, 2, \ldots$ until the optimal makespan is encountered

# SAT-based methods [Surynek 14]

SAT Encoding for bounded MAPF under makespan

- Define propositional variables $x^t_{j,k}$ s.t. $x^t_{j,k}$ is assigned true if agent $k$ appears at vertex $v_j$ at time step $t$
- Define constraints modeling validity conditions
  - An agent $k$ is placed in exactly one vertex at each time step $t$
  - At most one agent $k$ is placed in each vertex $j$ at each time step $t$
  - An agent $k$ relocates to some of its neighbors or makes no move at every time step $t$

$$x^t_{j,k} \Rightarrow \bigvee_{\ell:(v_j,v_\ell)\in E} x^{t+1}_{\ell,k} \qquad \text{for every time } t \text{ and every agent } k$$

$$x^{t+1}_{j,k} \Rightarrow \bigvee_{\ell:(v_\ell,v_j)\in E} x^{t}_{\ell,k} \qquad \text{for every time } t \text{ and every agent } k$$

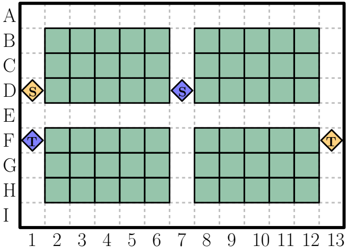
- Obtain a conjunctive normal form (CNF)
- Solve by trying for bounds $\mu = 1, 2, \ldots$ until the optimal makespan is encountered

# SAT-based methods [Surynek 14]

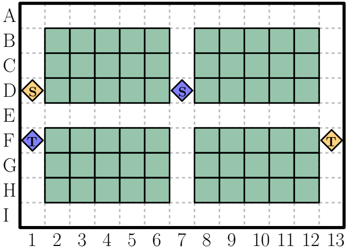
SAT Encoding for bounded MAPF under makespan

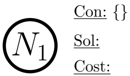
- Define propositional variables $x_{j,k}^t$ s.t. $x_{j,k}^t$ is assigned true if agent $k$ appears at vertex $v_j$ at time step $t$
- Define constraints modeling validity conditions
  - An agent $k$ is placed in exactly one vertex at each time step $t$
  - At most one agent $k$ is placed in each vertex $j$ at each time step $t$
  - An agent $k$ relocates to some of its neighbors or makes no move at every time step $t$
- Obtain a conjunctive normal form (CNF)
- Solve by trying for bounds $\mu = 1, 2, \ldots$ until the optimal makespan is encountered

# Conflict-based search [Sharon, Stern, Felner and Sturtevant 15]

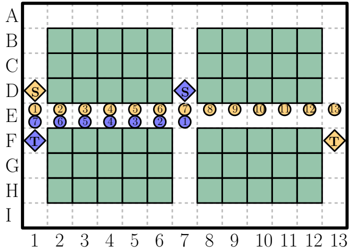


# Conflict-based search [Sharon, Stern, Felner and Sturtevant 15]



# Conflict-based search [Sharon, Stern, Felner and Sturtevant 15]



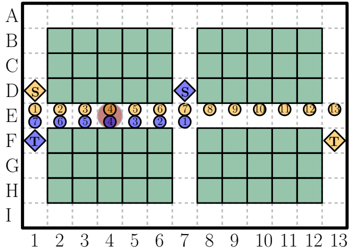


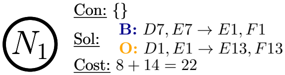
Con: {}

Sol: **B:** $D7, E7 \to E1, F1$
**O:** $D1, E1 \to E13, F13$

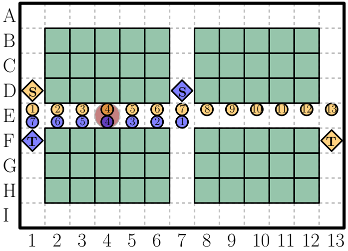
Cost: $8 + 14 = 22$

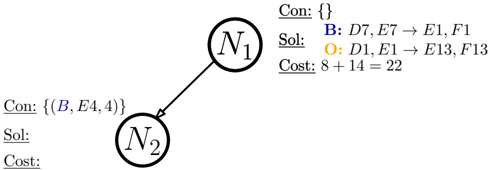
# Conflict-based search [Sharon, Stern, Felner and Sturtevant 15]



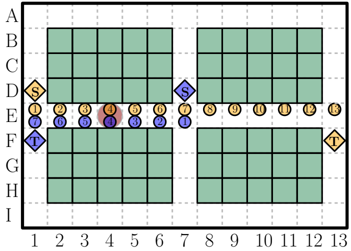



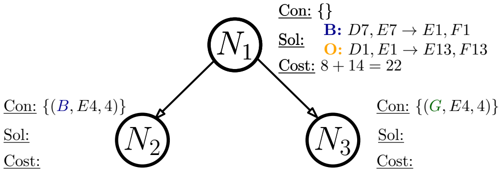
# Conflict-based search [Sharon, Stern, Felner and Sturtevant 15]



# Conflict-based search [Sharon, Stern, Felner and Sturtevant 15]



# Conflict-based search [Sharon, Stern, Felner and Sturtevant 15] (cont.)

- Since the exposition of CBS, heuristics have been used to push the running time of the algorithm by a factor of 50 [Li et al. 19]
- This has been done by
  - Choosing which conflicts to split on [[Boyarski et al.2015]
  - Heuristically guiding the constraint tree [Felner et al., 2018,Li et al. 19]
  - Relaxing optimality to bounded sub-optimality [Barer 14, Cohen et al. 16]
  - Applying domain knowledge such as adding corridors to the environment [Cohen, Uras and Koening 15]

# Discussion and alternative approaches

- Centralized vs. decentralized approaches
- Optimal vs. bounded sub-optimal approaches

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - "Clutterness" of environment?
  - "Tightness" of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?

  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - “Clutterness” of environment?
  - “Tightness” of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?
  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - “Clutterness” of environment?
  - “Tightness” of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?
  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - “Clutterness” of environment?
  - “Tightness” of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?
  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - "Clutterness" of environment?
  - "Tightness" of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?
  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - “Clutterness” of environment?
  - “Tightness” of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?

  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - “Clutterness” of environment?
  - “Tightness” of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?

  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - “Clutterness” of environment?
  - “Tightness” of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?
  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - “Clutterness” of environment?
  - “Tightness” of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?
  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization

# MAPF & MAPD—some research challenges

- What makes a MAPF problem computationally hard?
  - Number of robots ?
  - "Clutterness" of environment?
  - "Tightness" of passages in environment?
- How can we design effcient MAPF algorithms?
  - Pre-processing
- How can we design environments to maximize MAPD throughput?
  - Understand theoretic limitations
  - Use preprocessing
  - Design optimization