USC Viterbi School of Engineering

Search for Robotics: An Overview of Selected Research SOCS 2014

Sven Koenig
University of Southern California
idm-lab.org
skoenig@usc.edu

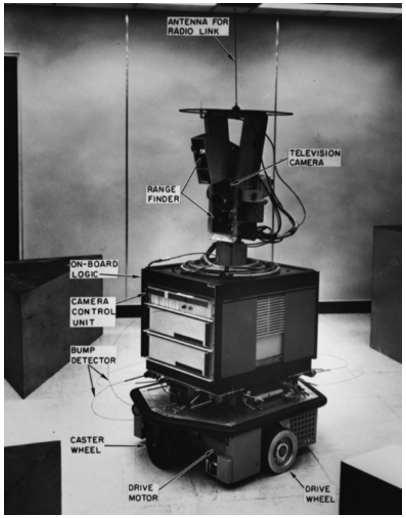
Search and Robotics

- E. Dijkstra, A Note on Two Problems in Connexion with Graphs, Numerische Mathematik 1, 269-271, 1959.
  "What is the shortest way to travel from Rotterdam to Groningen? It is the algorithm for the shortest path which I designed in about 20 minutes. One morning [in 1956] I was shopping with my young fiancée, and tired, we sat down on the café terrace to drink a cup of coffee and I was just thinking about whether I could do this, and I then designed the algorithm for the shortest path. … I designed it without pencil and paper."

  POP QUIZ
  So, one problem solved in the paper was the shortest path problem. What was the other problem?

Search and Robotics

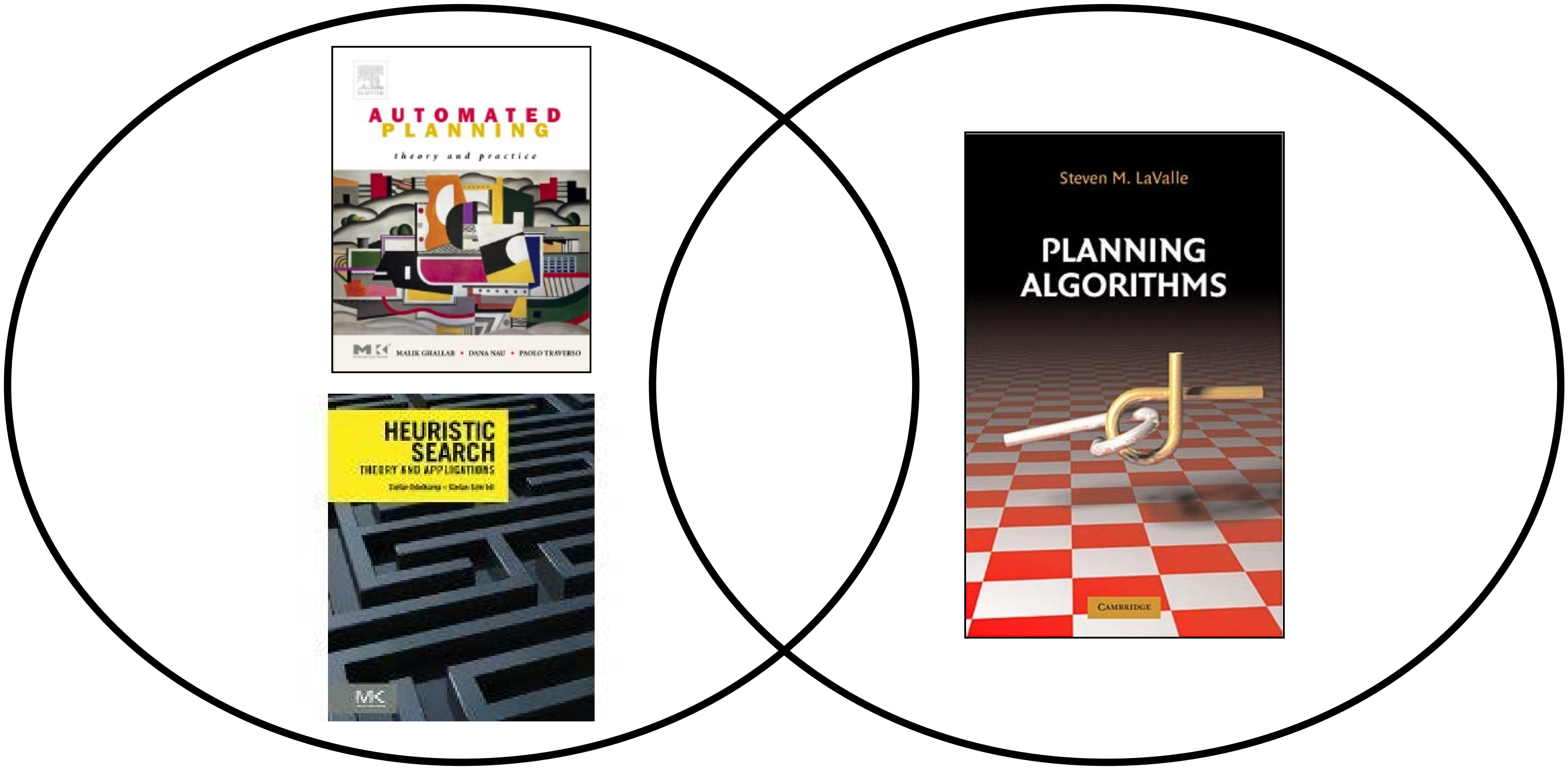
- P. Hart, N. Nilsson and B. Raphael, A Formal Basis for the Heuristic Determination of Minimum Cost Paths, IEEE Transactions on Systems Science and Cybernetics SSC4 4(2), 100-107, 1968.
- Shakey Project [1965(?)-1972]
  A*, STRIPS, visibility graphs, …



[Wikipedia]

POP QUIZ
What did
P. Hart, N. Nilsson and B. Raphael, Correction to "A Formal Basis for the Heuristic Determination of Minimum Cost Paths“, SIGART Newsletter 37: 28-29, 1972.
correct?

Search and Robotics



Artificial Intelligence

Robotics

Search and Robotics

- http://robotics.cs.unc.edu/PlanningWorkshop2013/

NSF-SPONSORED WORKSHOP:

Robot Planning in the Real World: Research Challenges and Opportunities

Overview | Location / Travel | Participants | Schedule / Slides | Report

Workshop Overview

In the last decade, researchers have made significant progress on robot planning, leading to impressive real-time planners for such challenging tasks as driving, flying, walking, and manipulating objects. Yet, robots are deployed in only a small number of niche areas, and most deployed robots have very minimal planning capability.

The aim of the workshop is to bring together people from academia, industry, and government research agencies to discuss how the field of robot planning should progress to make robots less reliant on human supervision and more widely deployable in the real world. The workshop brings together people with expertise in robotics, artificial intelligence, and related research disciplines to discuss the state of the art in planning, its use in various robotic applications, and current research challenges. By studying planning research across different applications, analyzing planning challenges as part of complete robot architectures, and discussing the interaction of planning with other robot modules (such as perception, control, and user interfaces), the workshop participants will gain new insights into how planning can help robots become more robust and efficient. The workshop consists of invited talks, breakout sessions, panels, and a final discussion to create a roadmap for the field of robot planning that will be summarized in a report. The workshop and the resulting report will have the potential to stimulate future research on robot planning across many applications, from personal assistance to medicine to exploration to manufacturing.

Search and Robotics

- SOCS 2013: about 30 accepted papers

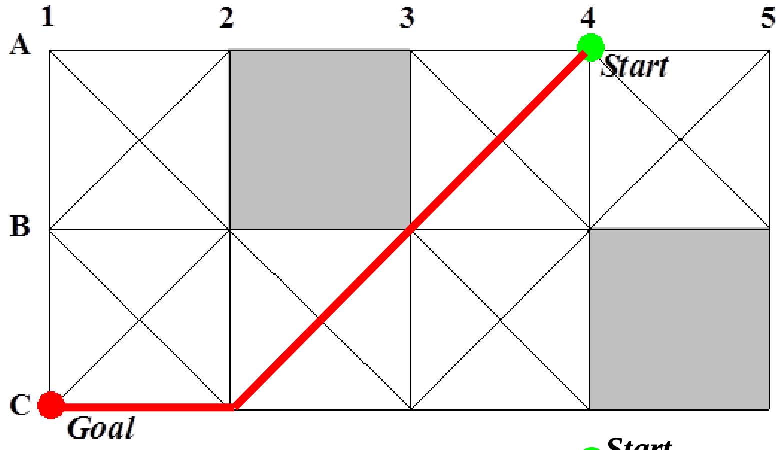
- AAAI 2013: about 250 accepted papers
- IJCAI 2013: about 490 accepted papers

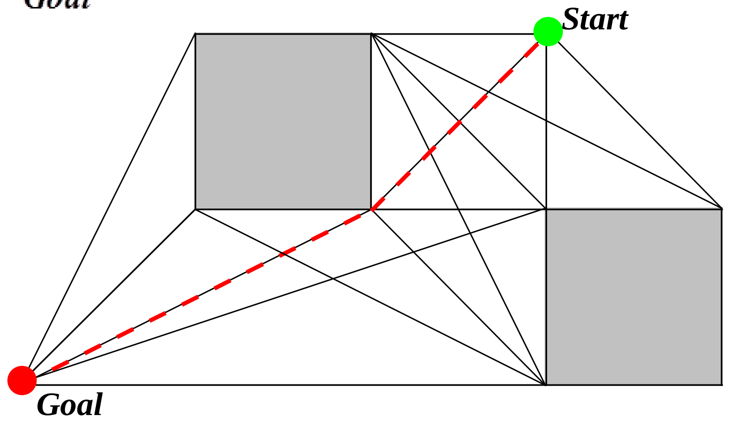
- ICRA 2013: about 870 accepted papers
- IROS 2013: about 900 accepted papers

TOC

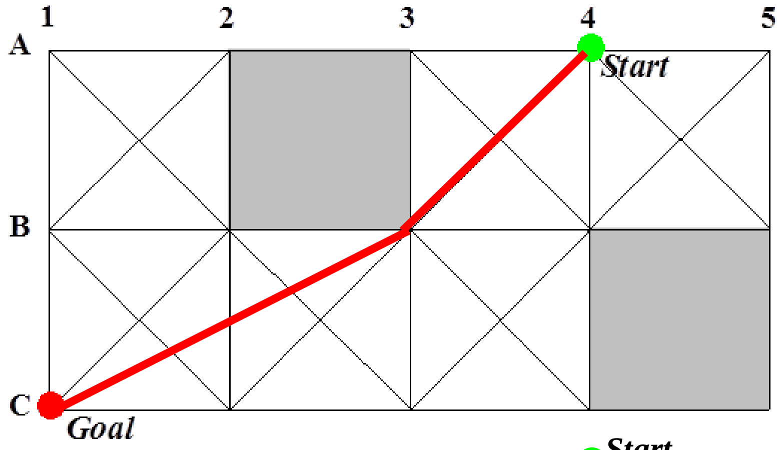
- Search and Robotics
  - Overview
  - Problem Classes: Examples of Current Research
    - Any-Angle Search
    - Search for Motion Planning
    - Multi-Robot Path Planning
    - Replanning
    - Plan Reuse
  - Specific Robot Task Planning Problem: “Blocks World”

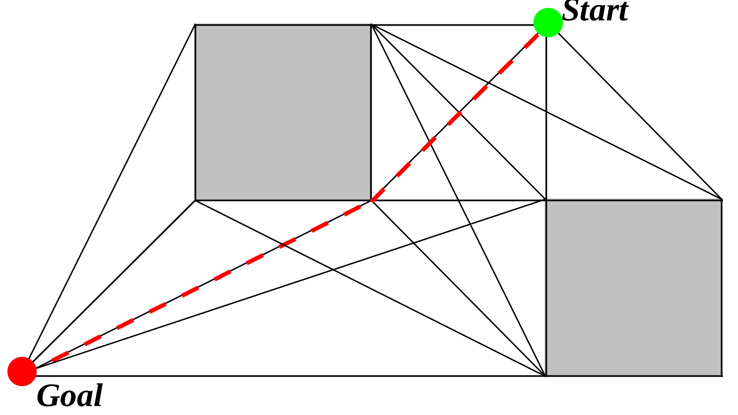
Any-Angle Search





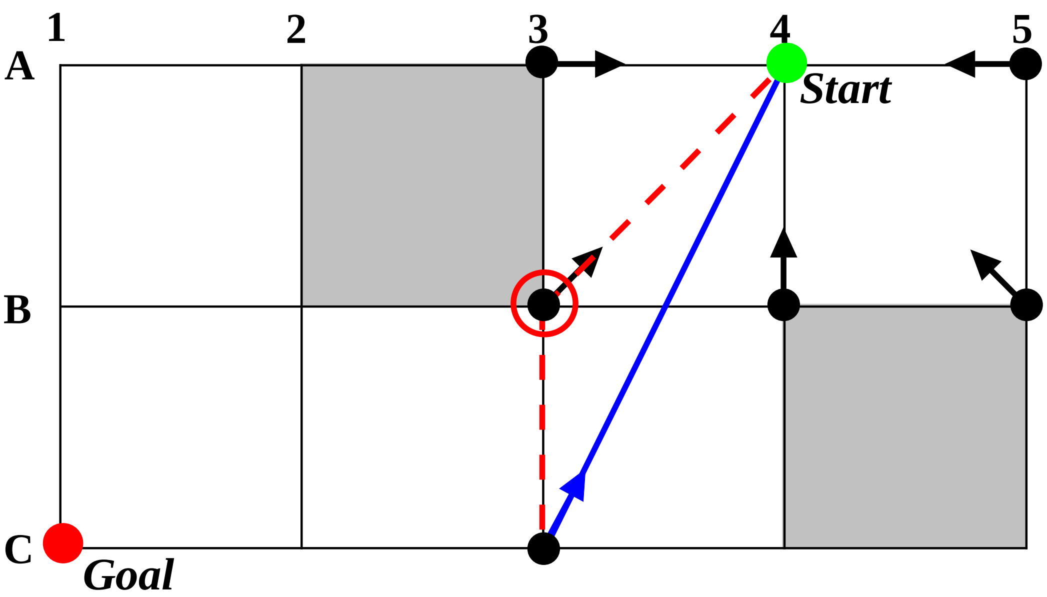
Any-Angle Search





Any-Angle Search

- How much shorter can the shortest any-angle paths be than the shortest grid paths?
- How to find short any-angle paths fast?



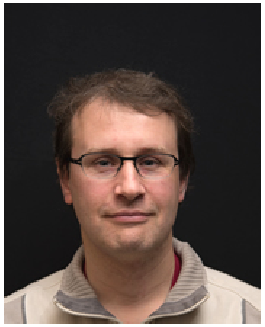
Theta* by A. Nash + collaborators

[JPL]

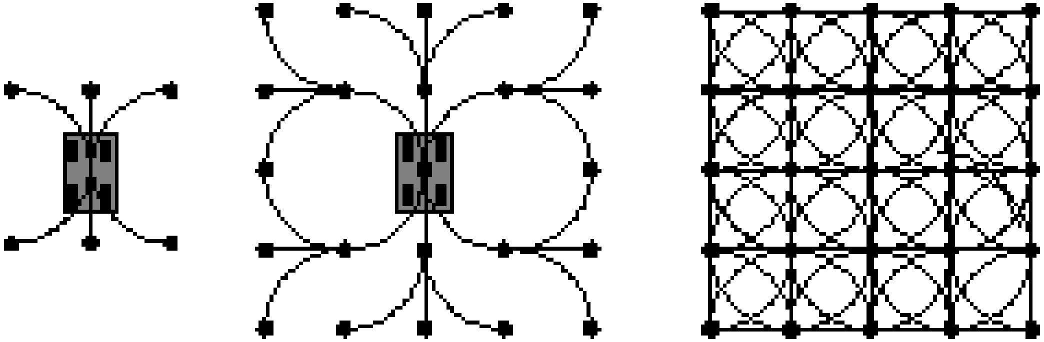
Field D* by D. Ferguson + collaborators

[blog.wcgworld.com]

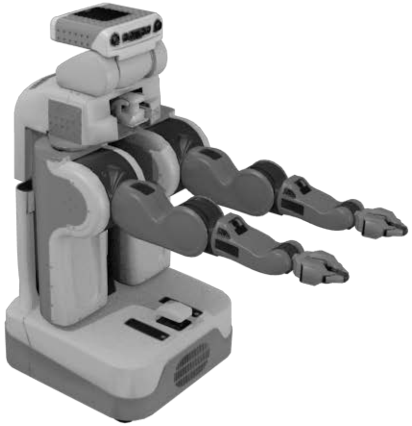
Search for Motion Planning



- Search-Based Planning Library by M. Likhachev + collaborators



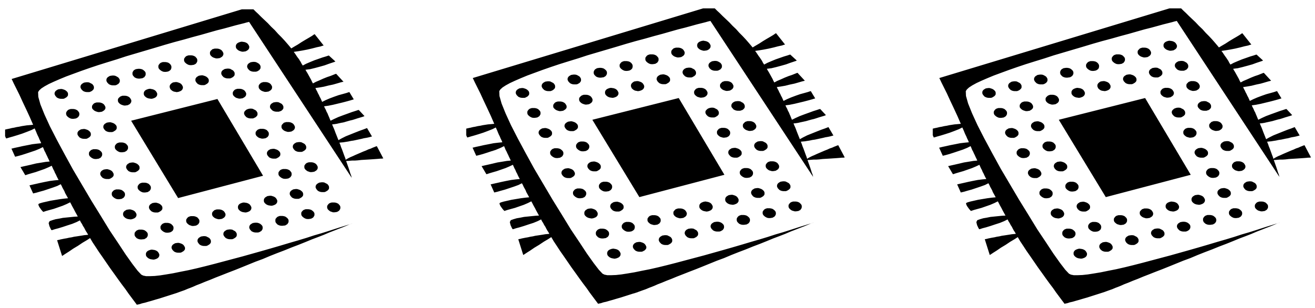
[Mikhail Pivtoraiko]



[openrobots.org]

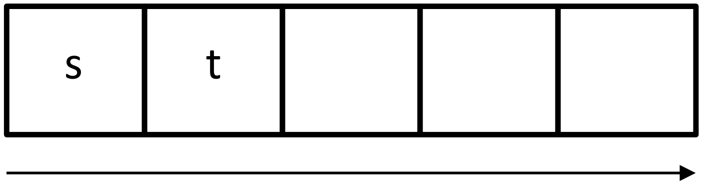
- Node expansions are time consuming
- State spaces are high-dimensional (= large)
- There is lots of structure that can be exploited
  - For hierarchical planning
  - For receding horizon planning
  - For calculating heuristics

Parallel Search



Parallel Search

Parallel A* for Slow Expansions by M. Phillips + collaborators



- f(s) = 5, g(s) = 1
- f(t) = 7, g(t) = 3
- We need to guard against g(t) > g(s) + c*(s,t)

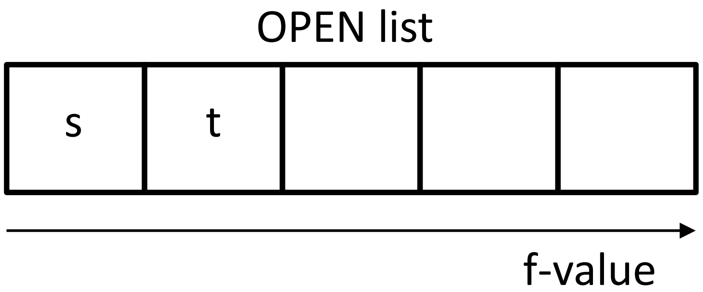
s g(s) = 1

c*(s,t) = 1

t g(t) = 3

Parallel Search

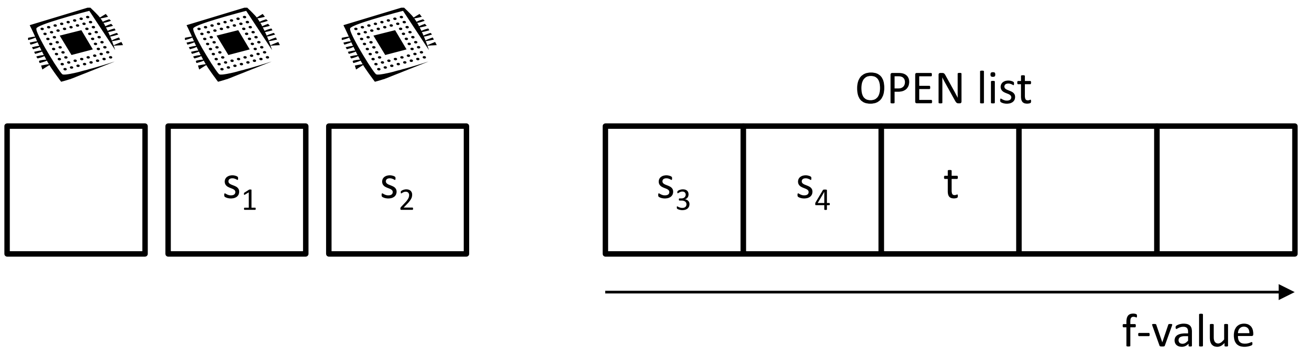
Parallel A* for Slow Expansions by M. Phillips + collaborators



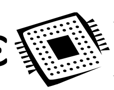
- $f(s) = 5$, $g(s) = 1$
- $f(t) = 7$, $g(t) = 3$
- We need to guard against $g(t) > g(s) + c^*(s,t)$
- Expand t only if $g(t) \leq g(s) + h(s,t)$
  because then $g(t) \leq g(s) + h(s,t) \leq g(s) + c^*(s,t)$

Parallel Search

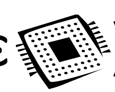
Parallel A* for Slow Expansions by M. Phillips + collaborators



A*

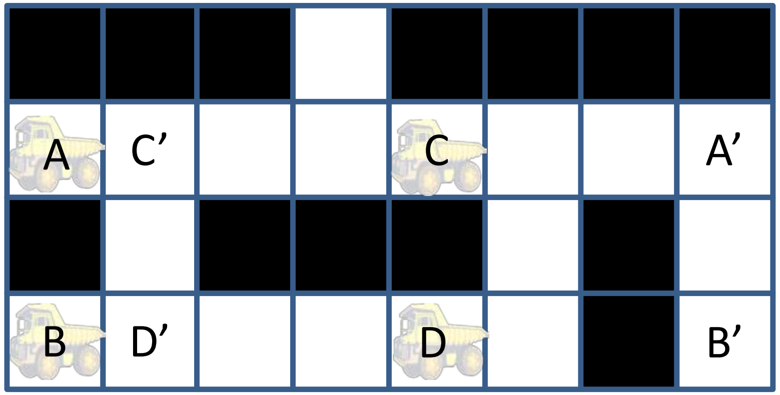
- $g(t) \le g(s_i) + h(s_i,t)$ for all ($s_i \in$ ) or ($s_i \in$ OPEN with $f(s_i) < f(t)$)

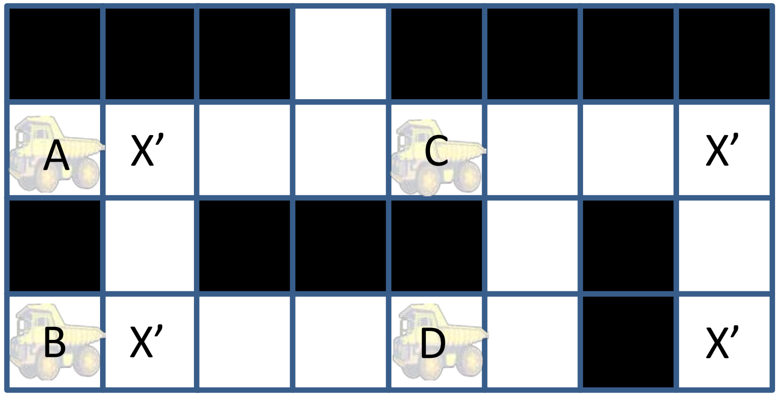
Weighted A* with weight $w \ge 1$ and bound $w' \ge w$

- $g(t) \le g(s_i) + w' h(s_i,t)$ for all ($s_i \in$ ) or ($s_i \in$ OPEN with $f(s_i) < f(t)$)

Multi-Robot Path Planning

- AAAI-12 Workshop on Multi-Agent Pathfinding





Get robot A to A’
Get robot B to B’
Get robot C to C’
Get robot D to D’

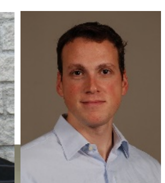
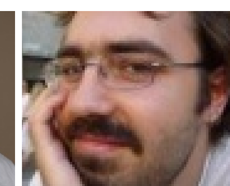
Get robot A to one of the X’
Get robot B to one of the X’
Get robot C to one of the X’
Get robot D to one of the X’
so that all of the X’ are visited

Multi-Robot Path Planning

- How complex is optimal and non-optimal multi-robot path planning?
- How to plan fast?

[www.ilex-press.com]

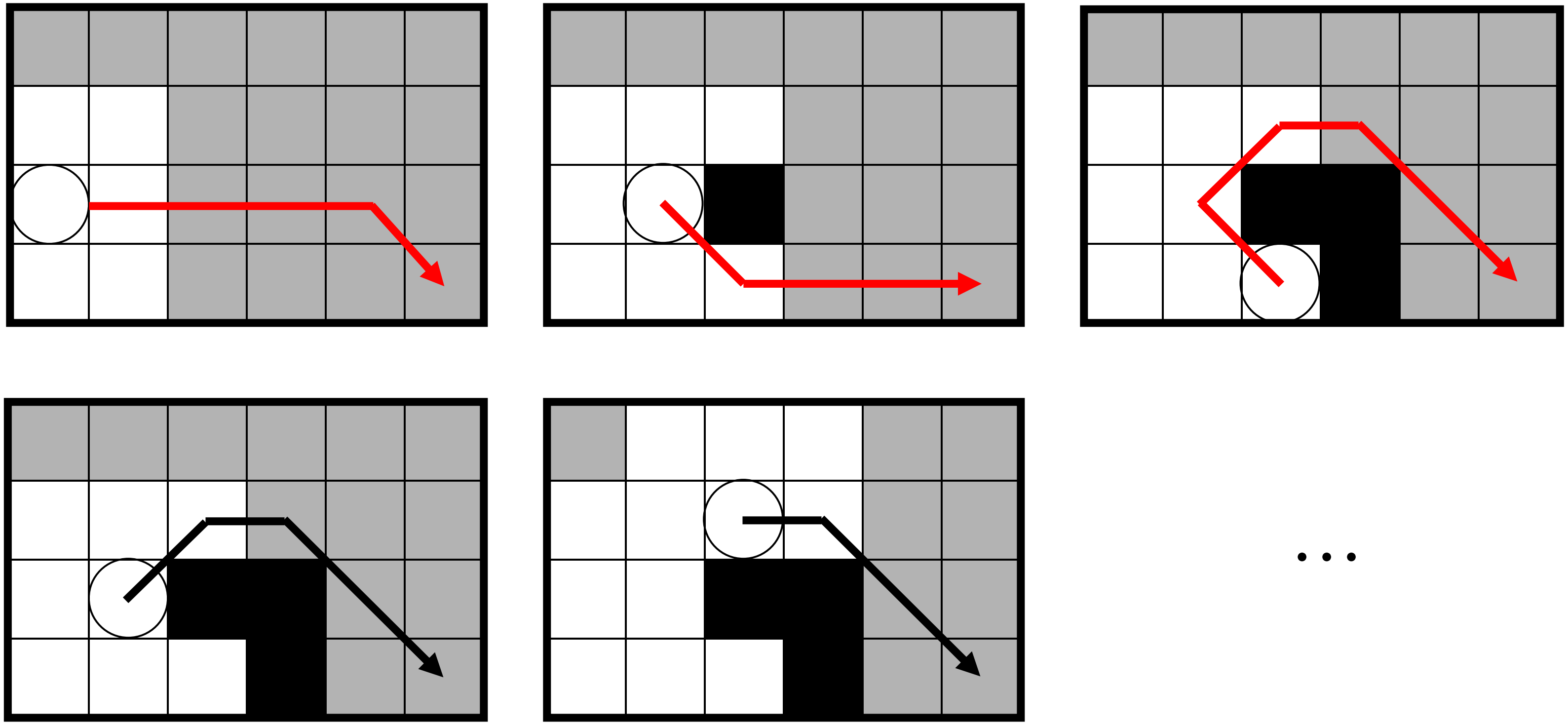
[www.easternlifttruck.co


[Örebro University]

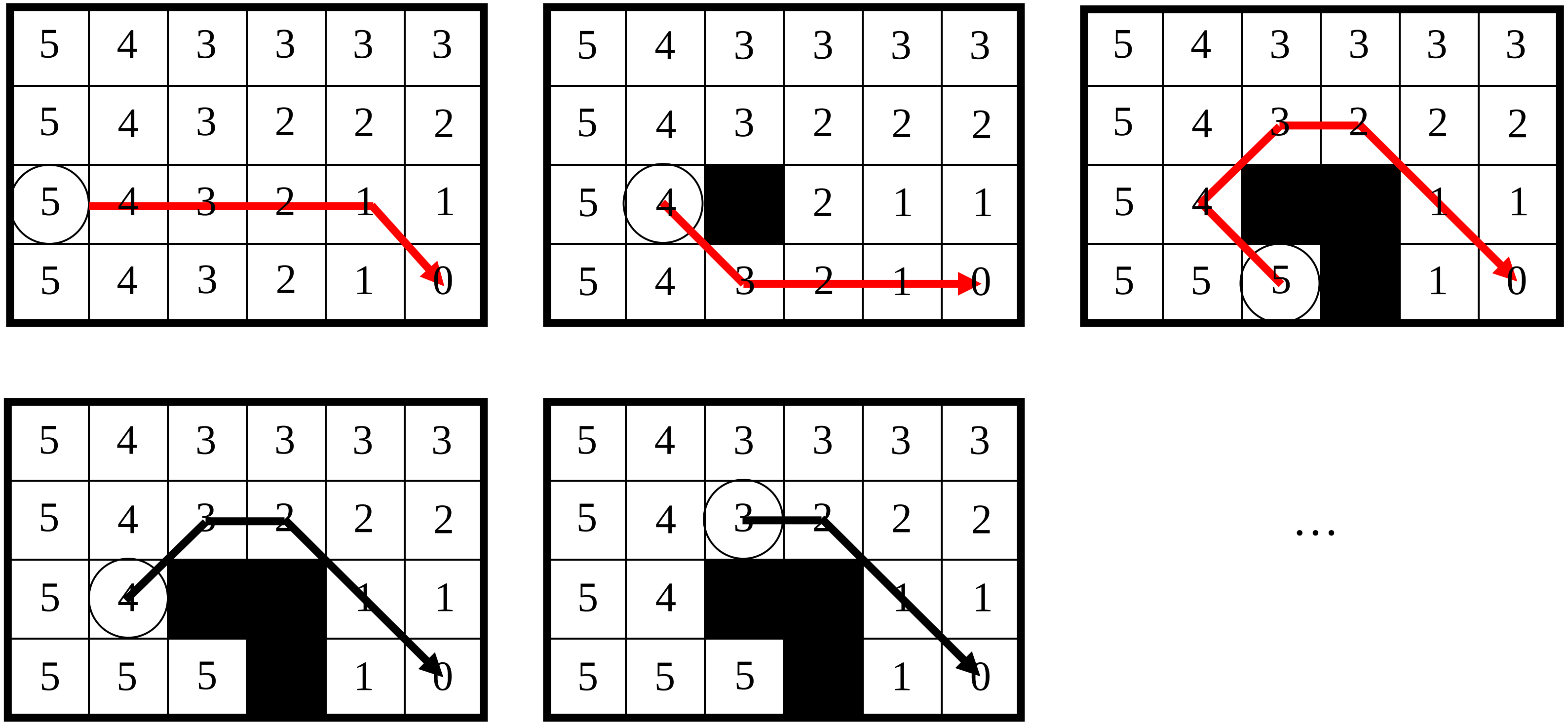
Replanning

- Incremental heuristic search



Replanning

- Incremental heuristic search



Replanning

- How long is the resulting trajectory at most?
- How to replan fast?
  - By restarting the search
  - By transforming the old search tree to the new search tree
  - By making the heuristic values more informed

[Carnegie Mellon University]

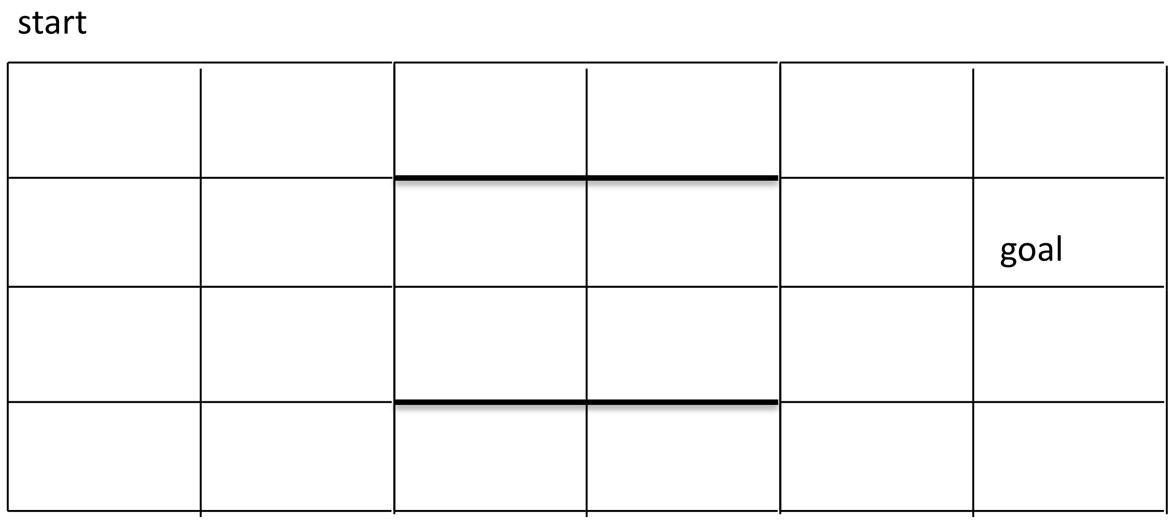
Plan Reuse

- Experience graphs by Mike Phillips + collaborators

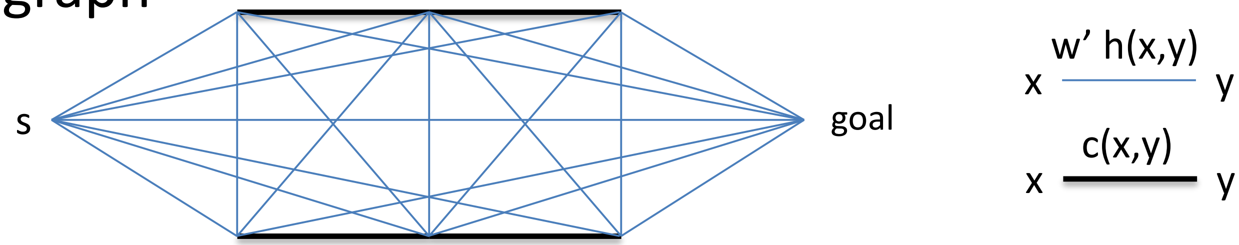
[homeguides.sfgate.com]

Plan Reuse

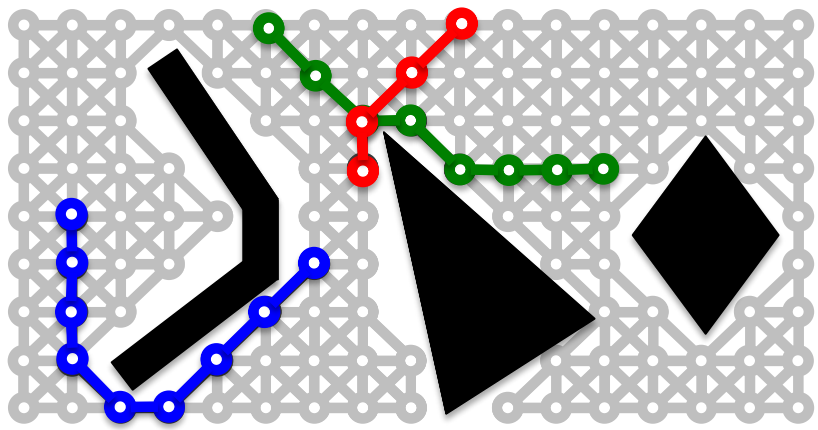
- Perform a weighted A* search with weight $w \geq 1$, resulting in bound $w \times w'$



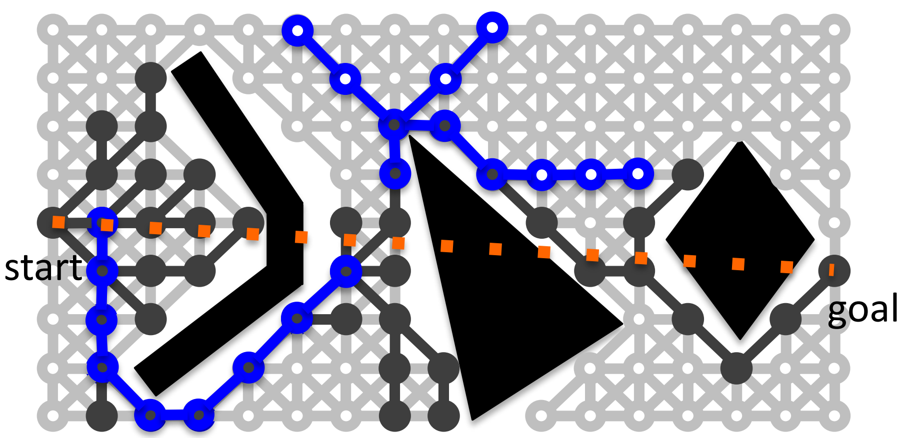
- Calculate the heuristic $h(s)$ of any state s as its goal distance on the following graph



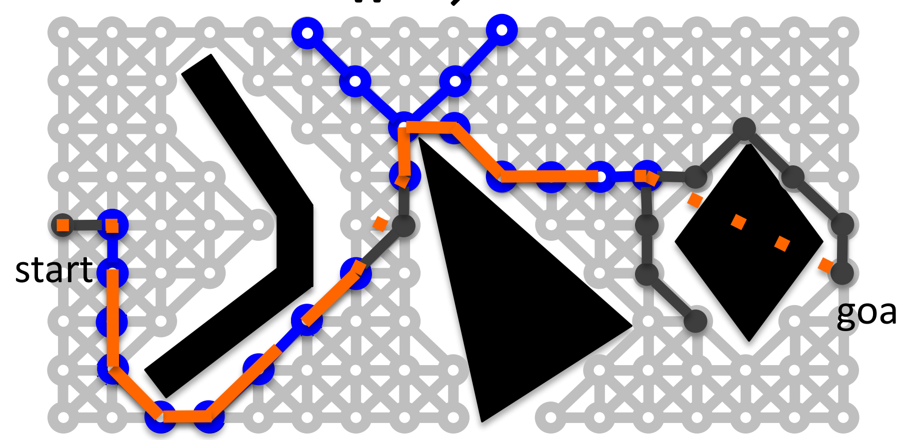
Plan Reuse



w’ = 1.5



w’ → ∞



[Mike Phillips]

Other Topics

Many other interesting topics

- Time-sensitive search
- Combining task and motion planning
- …

TOC

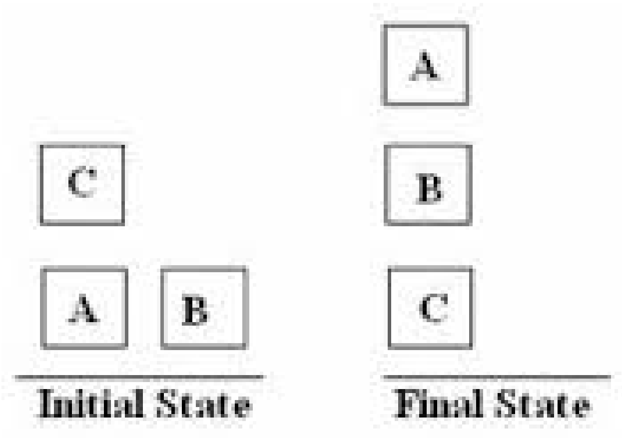
- Search and Robotics
  - Overview
  - Problem Classes: Examples of Current Research
  - Specific Robot Task Planning Problem: “Blocks World” - A Tree-Based Algorithm for Construction Robots
    by S. Kumar, S. Jung + collaborators

Search in Robotics

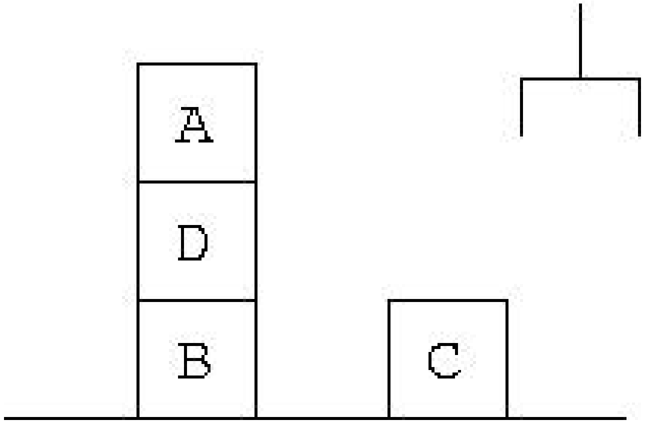
- Search in robotics is complicated due to
  - Continuous state spaces
  - Uncertainty
  - Kinematic and dynamic constraints
  - Combinations of task and motion planning

- What if we are looking for a simpler start?

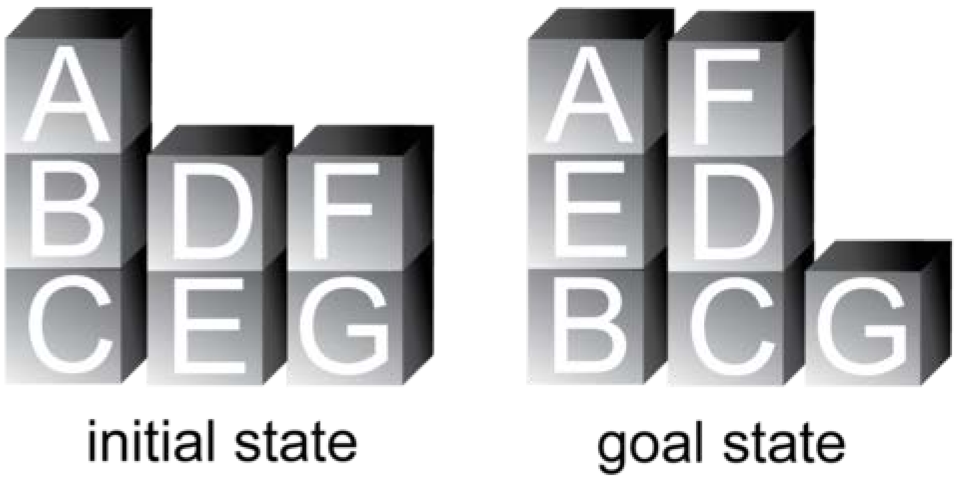
Blocks World



[iete-elan.ac.in]



[www.cs.bham.ac.uk]



[en.wikipedia.org]

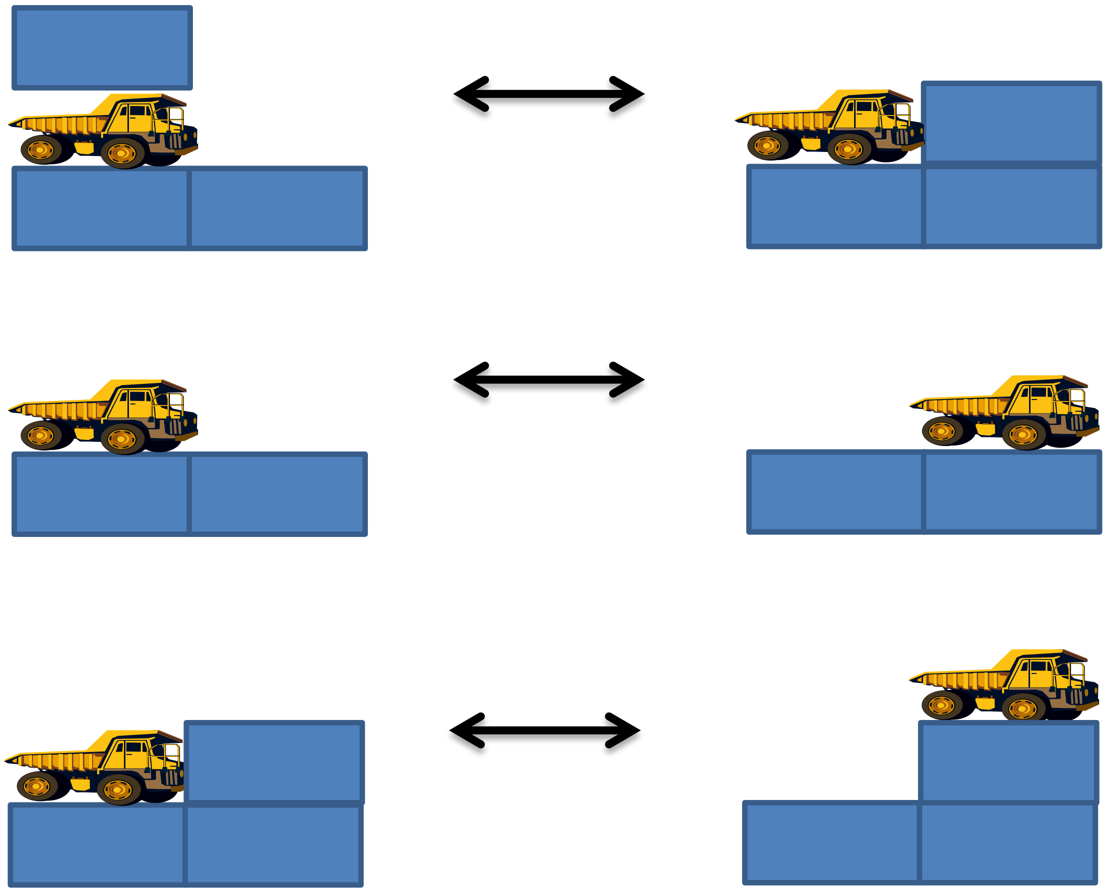
Harvard TERMES Project

- TERMES project by Radhika Nagpal + collaborators

[Harvard University]

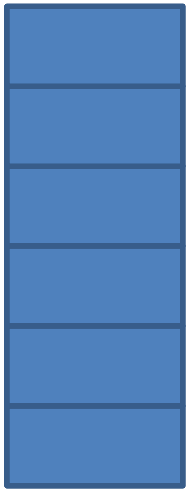
Construction Robots

- Capabilities of the robots



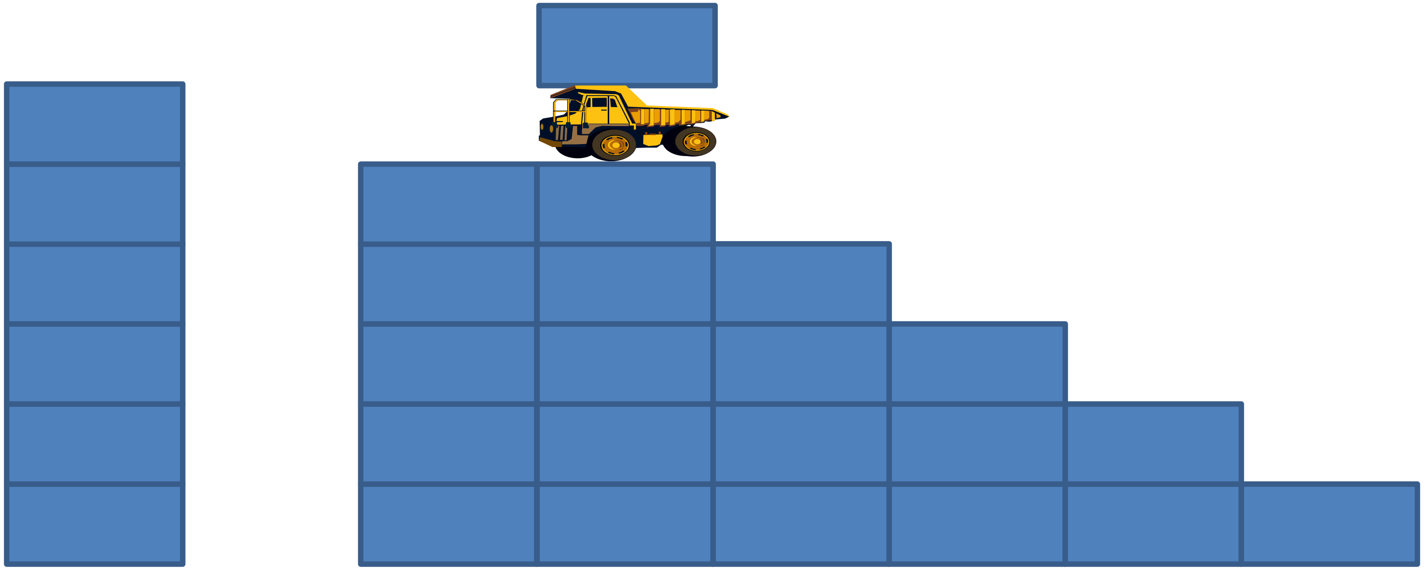
Construction Robots

- Difficulty



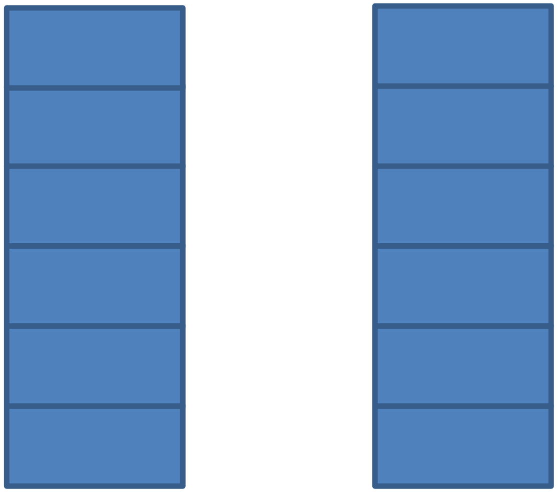
Construction Robots

- Difficulty



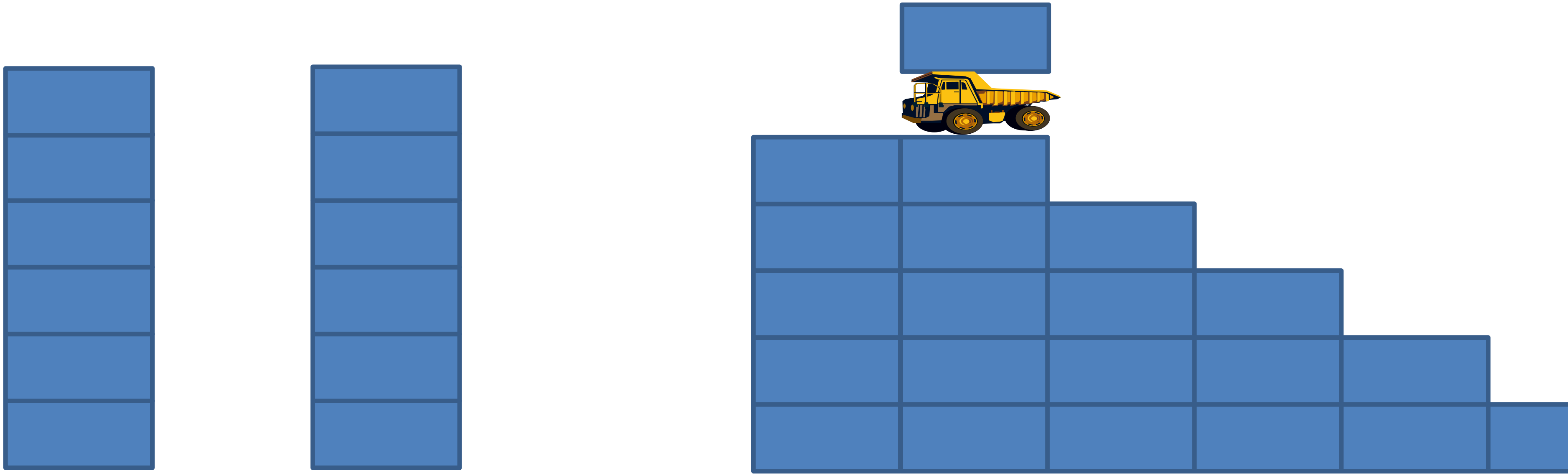
Construction Robots

- Difficulty



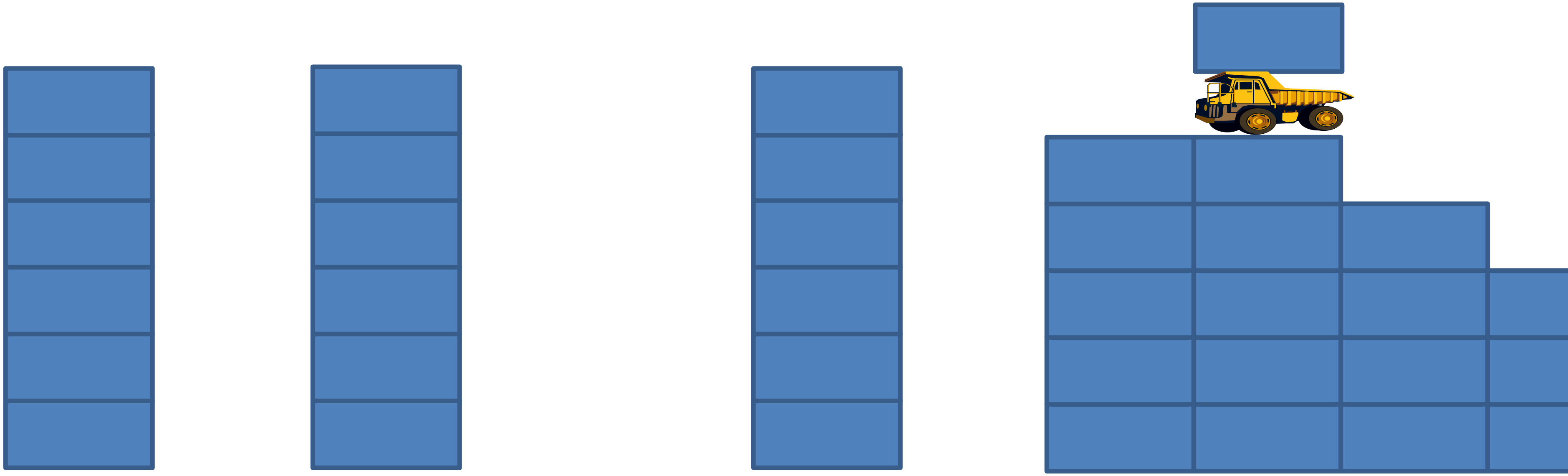
Construction Robots

- Difficulty



Construction Robots

- Difficulty



Construction Robots

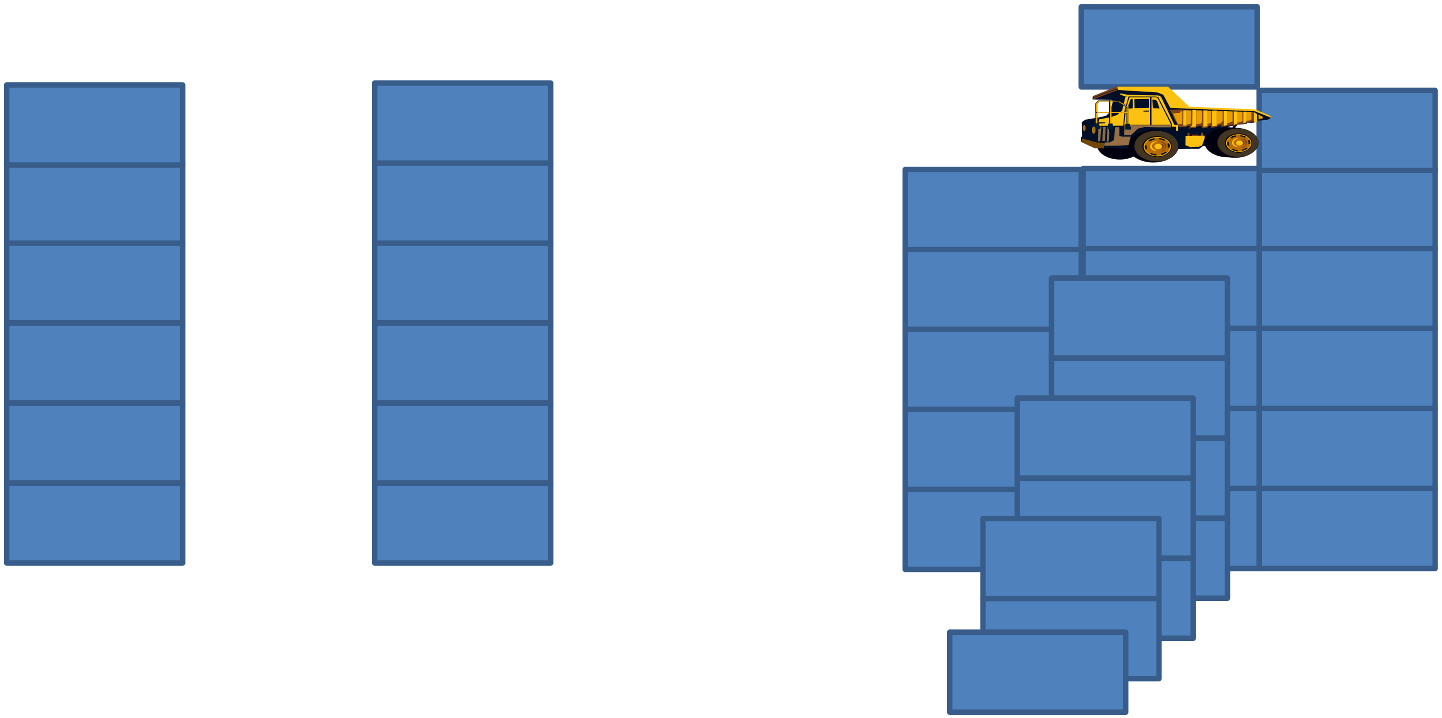
- Tree-based dynamic programming

1	1	1	1	1
1				1
1		3		1
1				1
1	1	1	1	1

Tower by Tower (TBT) Method

Construction Robots

- Difficulty



Construction Robots

- Difficulty
  - Behavior-based robotics?
  - General-purpose planning?

- We developed a special-purpose planner

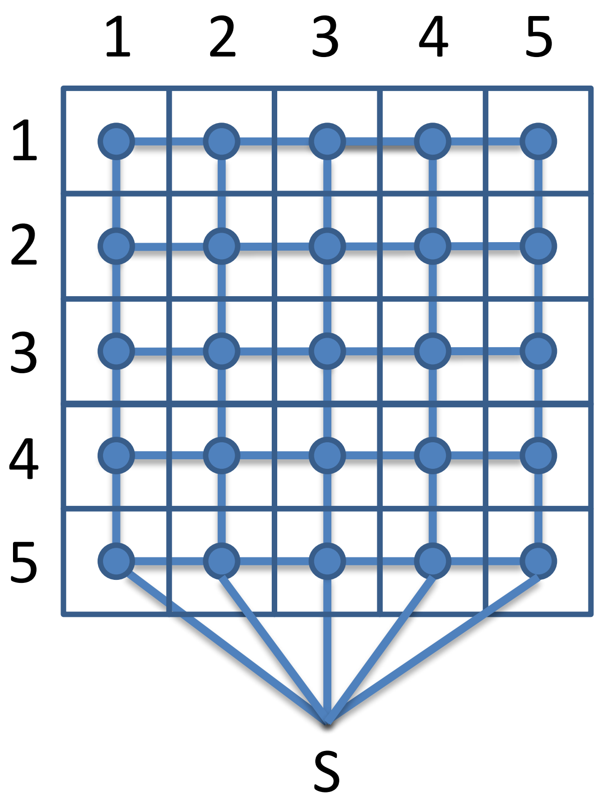
Construction Robots

- Tree-based dynamic programming

1	1	1	1	1
1				1
1		3		1
1				1
1	1	1	1	1

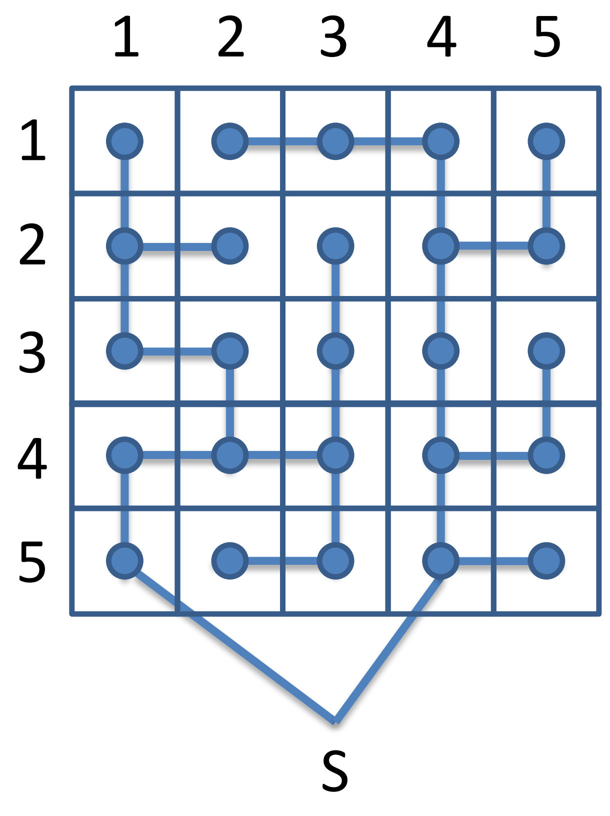
Construction Robots

- Tree-based dynamic programming



Construction Robots

- Tree-based dynamic programming

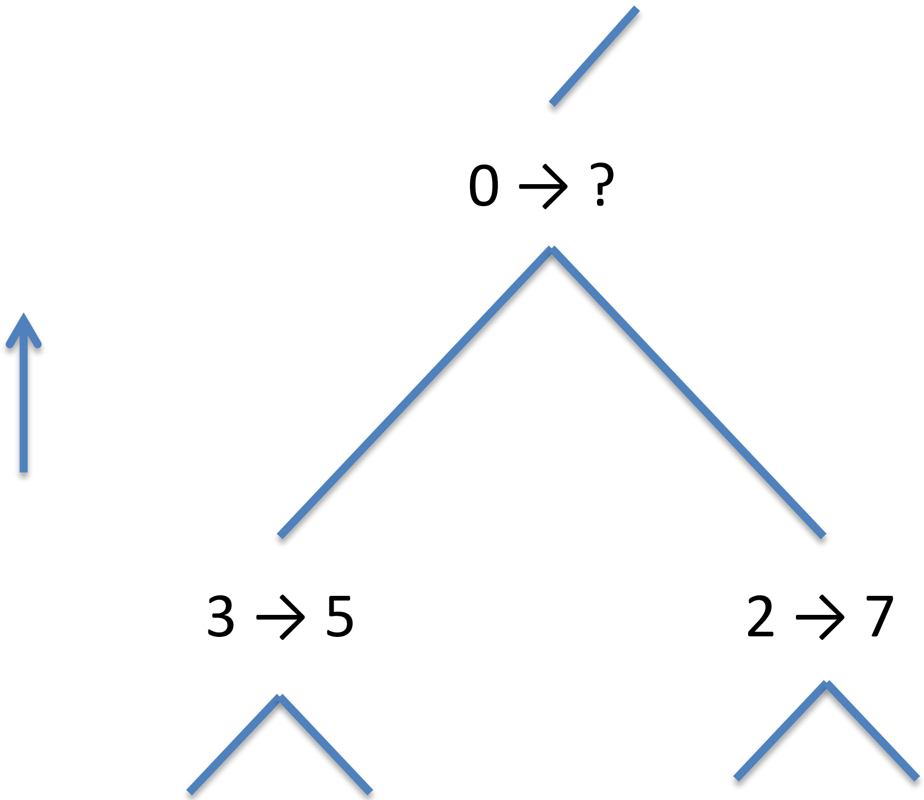


Construction Robots

Put down blocks	Pick up blocks	Put down blocks	…

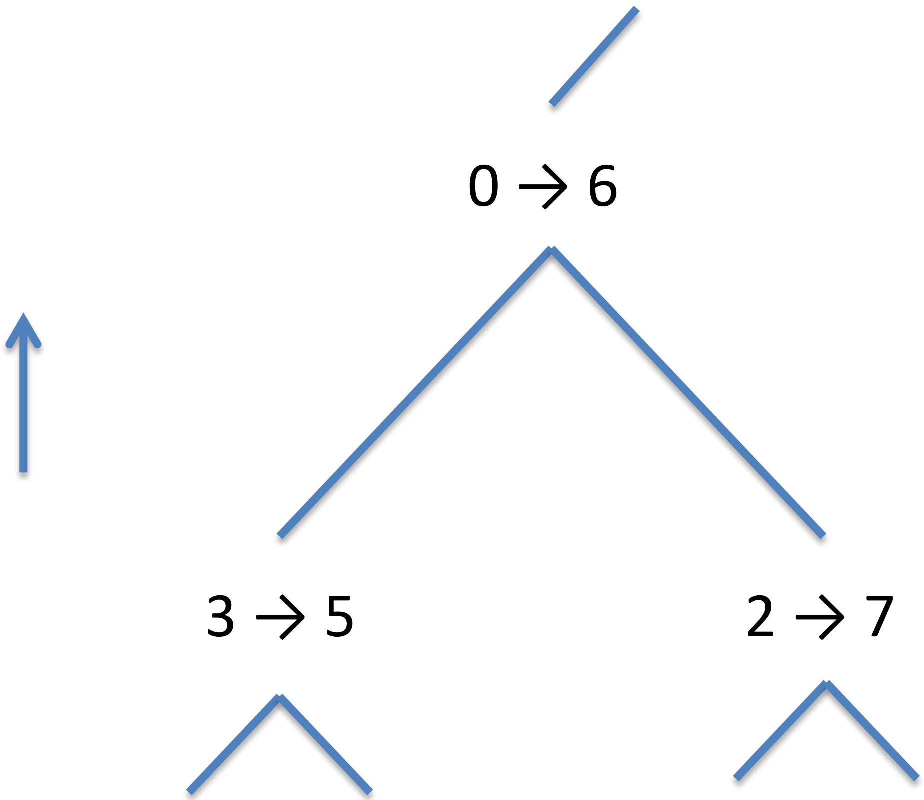
Single Robot Case

- Tree-based dynamic programming



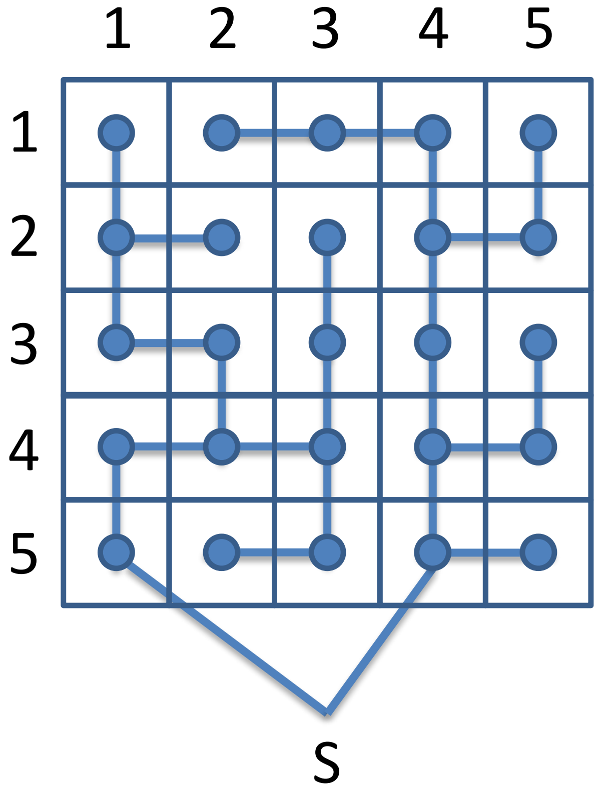
Single Robot Case

- Tree-based dynamic programming



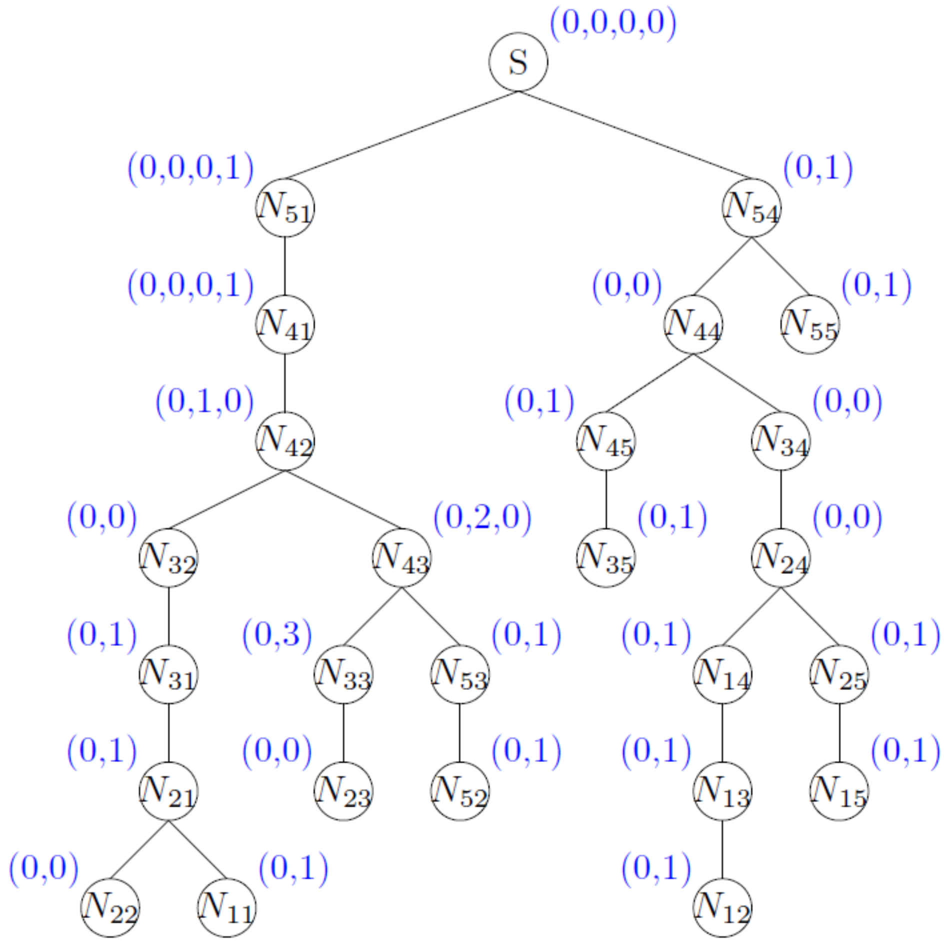
Construction Robots

- Tree-based dynamic programming



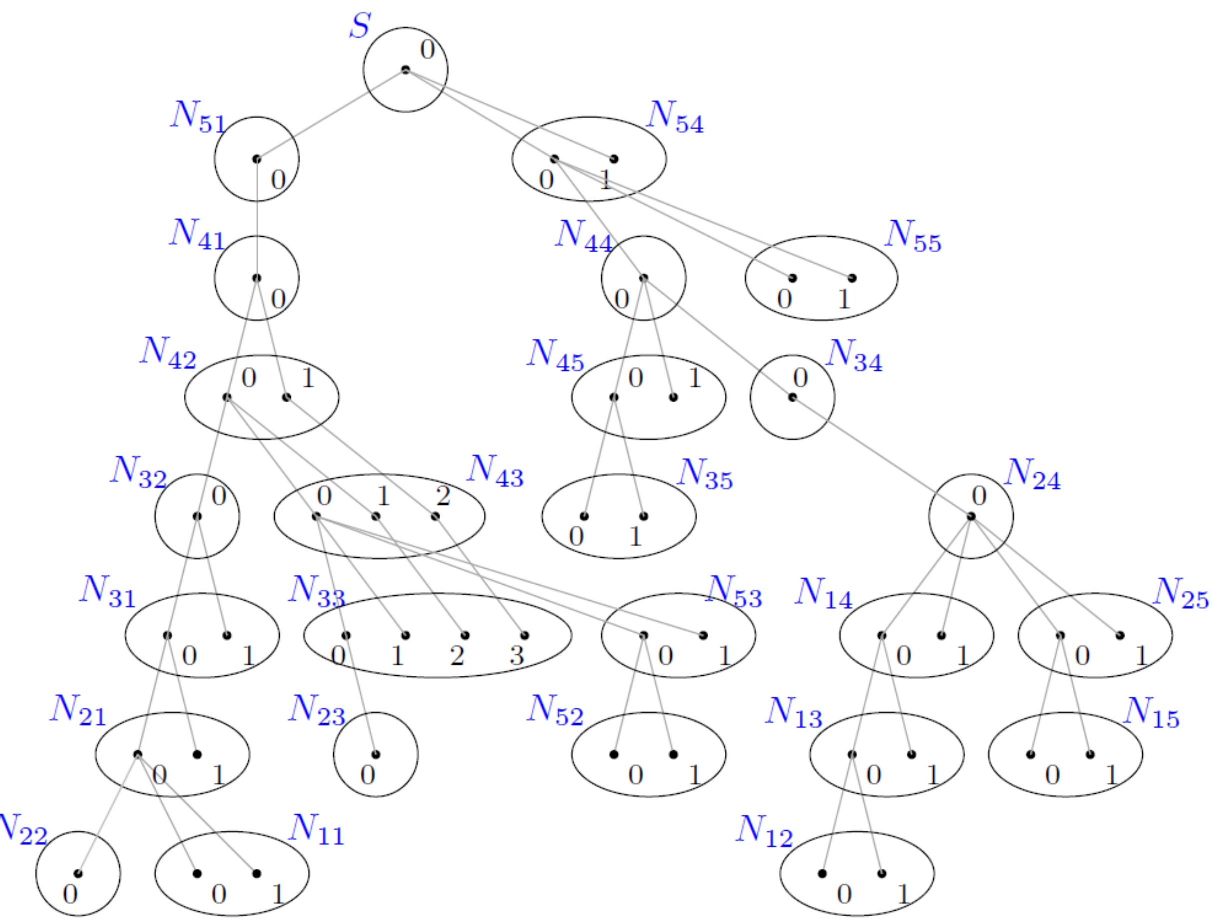
Single Robot Case

- Tree-based dynamic programming



Single Robot Case

- Tree-based dynamic programming



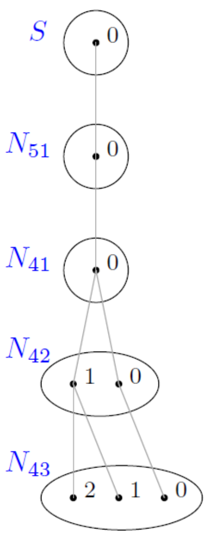
1	1	1	1	1
1				1
1		3		1
	1	2		1
	1	1	1	1

Positive event tree: add blocks

The support of a node with value ***v*** is that node in the parent super-node with the lowest value greater than or equal to ***v* – *1***.

Single Robot Case

- Tree-based dynamic programming



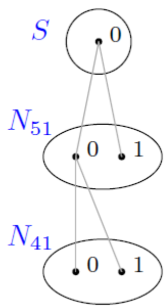
1	1	1	1	1
1				1
1		3		1
				1
	1	1	1	1

Negative event tree: remove blocks

The support of a node with value ***v*** is that node in the parent super-node with the highest value smaller than or equal to ***v.***

Single Robot Case

- Tree-based dynamic programming



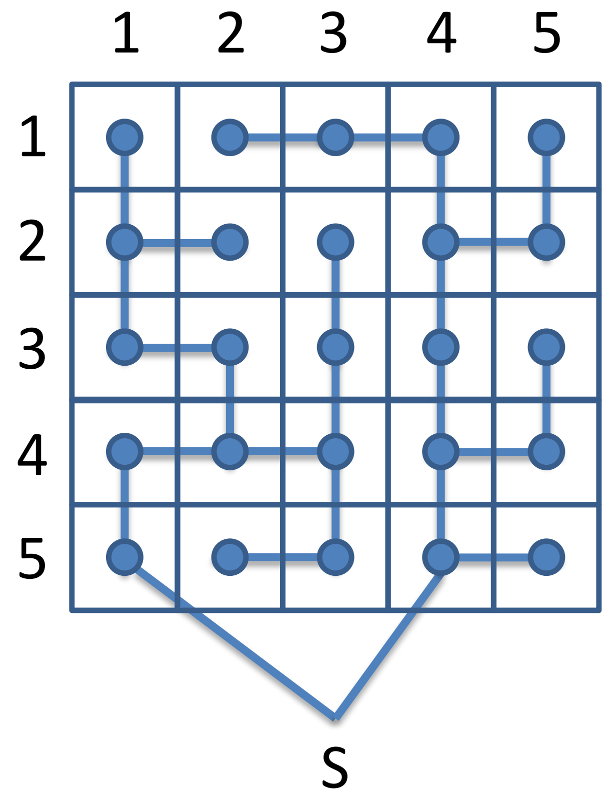
1	1	1	1	1
1				1
1		3		1
1				1
1	1	1	1	1

Positive event tree:
add blocks

The support of a node with value ***v*** is that node in the parent super-node with the lowest value greater than or equal to ***v – 1***.

Construction Robots

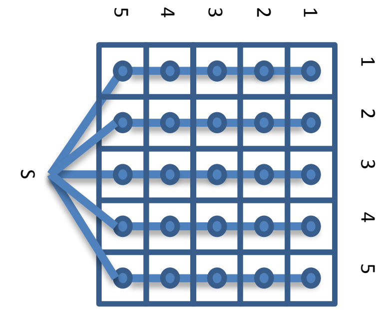
- Tree-based dynamic programming



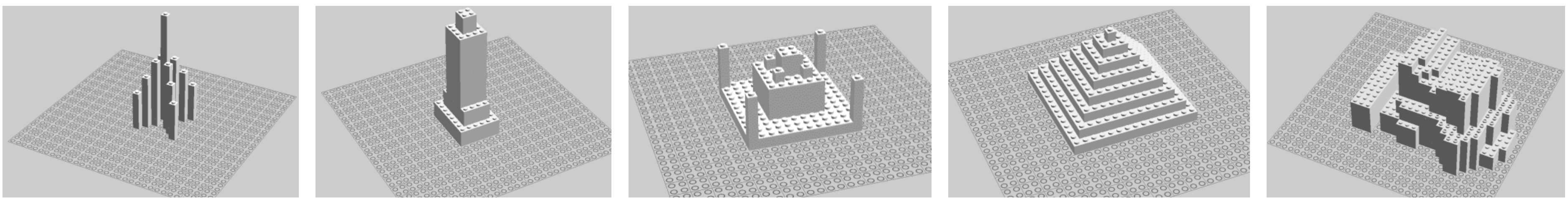
Single Robot Case

- A good spanning tree is crucial.
- Find a minimum spanning tree:
  - Edge from S have weight 0.
  - Edges between cells have cost = their height difference.

Single Robot Case



Row by Row (RBR) Method



[source unknown]

Building Model	Matrix	Max H	TBT	RBR	MST
Eiffel Tower	7×7	15	845	845	781
Empire State	6×8	15	3152	932	450
Taj Mahal	12×12	6	896	384	352
Giza Pyramid	15×15	8	2752	680	680
Disney Hall	22×16	10	11091	2245	1493

Number of block operations

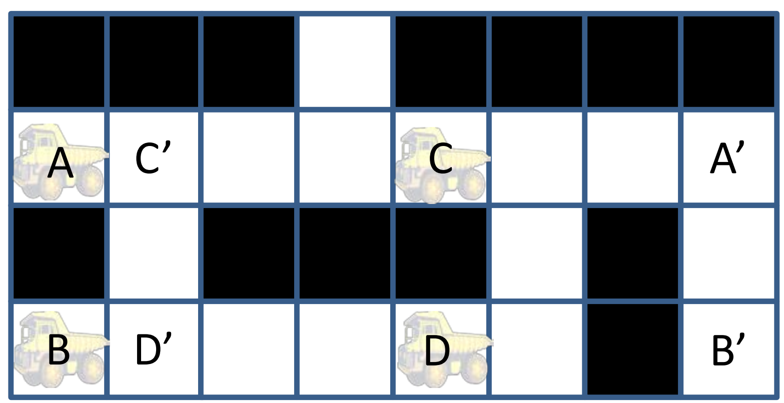
TBT = Tower by Tower Method
RBR = Row by Row Method
MST = Minimum Spanning Tree Method (described on the previous slides)

Single Robot Case

- Ongoing work
  - Different objective functions
  - Local search to determine the spanning tree
  - Hollow structures with roofs
  - Non-uniform block sizes

Multi-Robot Case

- Ongoing work
  - Spanning trees allow for parallelism since different robots might be able to work on different subtrees.
  - Two robots cannot pass each other on a ramp. Thus, one needs to solve a multi-robot path planning problem.



Multi-Robot Case

- Ongoing work
  - Spanning trees allow for parallelism since different robots might be able to work on different subtrees.
  - Two robots cannot pass each other on a ramp. Thus, one needs to solve a multi-robot path planning problem.
  - Multiple robots can implement strategies that single robots cannot implement, for example, bucket brigades.

Conclusions

- This talk reports on research from a large number of colleagues and students, some of which are not our collaborators. Many thanks to Maxim Likhachev's research group, where I spent part of my sabbatical in 2013/14!
- Find our papers at:
  idm-lab.org.
- Please talk to me or send me an email if you have any good ideas related to any of these topics:
  Sven Koenig – skoenig@usc.edu.
- We acknowledge funding from:
  NSF, ARO and ONR.

Thank you!