NBS applied to Planning

Marvin Buff

University of Basel

11. February, 2019

Motivation

Foundation

Classical Planning: Devise a new efficient and competitive algorithm.

Inspiration

The introduction of a new and promising bidirectional search algorithm: *Near-Optimal Bidirectional Search Algorithm (NBS).*

Realization

- Implementing NBS as planner.
- Evaluating its performance on the IPC benchmark.
- Analysing the results with fMM.

Motivation

Foundation

Classical Planning: Devise a new efficient and competitive algorithm.

Inspiration

The introduction of a new and promising bidirectional search algorithm: *Near-Optimal Bidirectional Search Algorithm (NBS)*.

Realization

- Implementing NBS as planner.
- Evaluating its performance on the IPC benchmark.
- Analysing the results with fMM.

Motivation

Foundation

Classical Planning: Devise a new efficient and competitive algorithm.

Inspiration

The introduction of a new and promising bidirectional search algorithm: *Near-Optimal Bidirectional Search Algorithm (NBS)*.

Realization

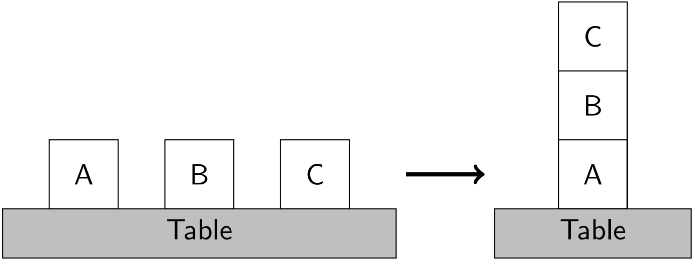
- Implementing NBS as planner.
- Evaluating its performance on the IPC benchmark.
- Analysing the results with fMM.

Outline

Outline

NBS

The Blocks Domain

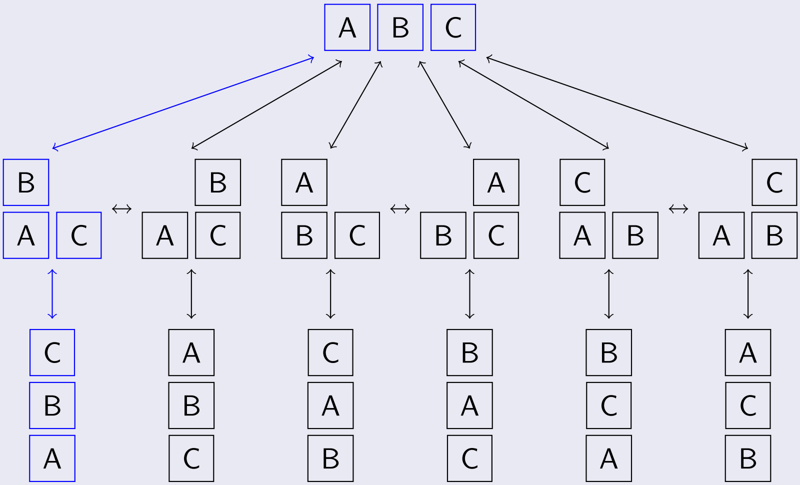


Actions

- Pick Up
- Put Down

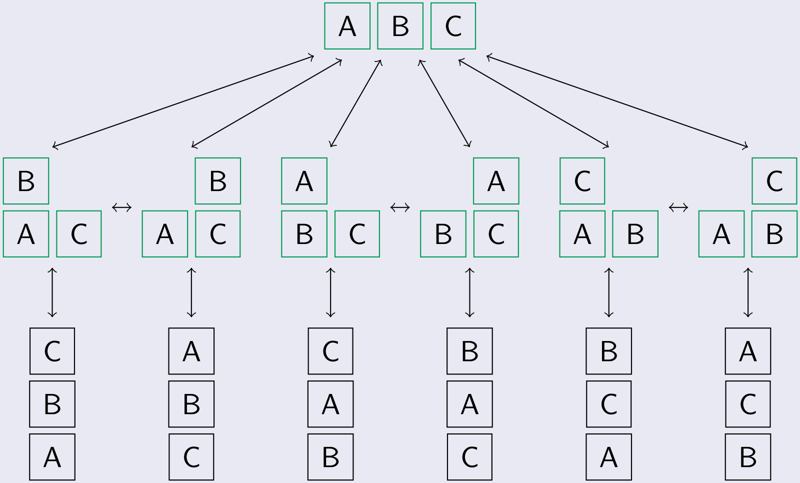
The Blocks Domain

Transition System



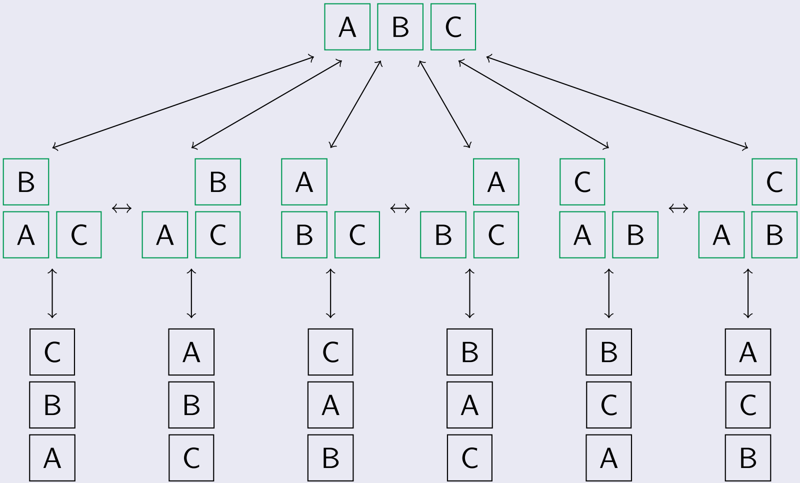
Sufficient Conditions

Surely Expanded States: $f(s) < C^* = 2$



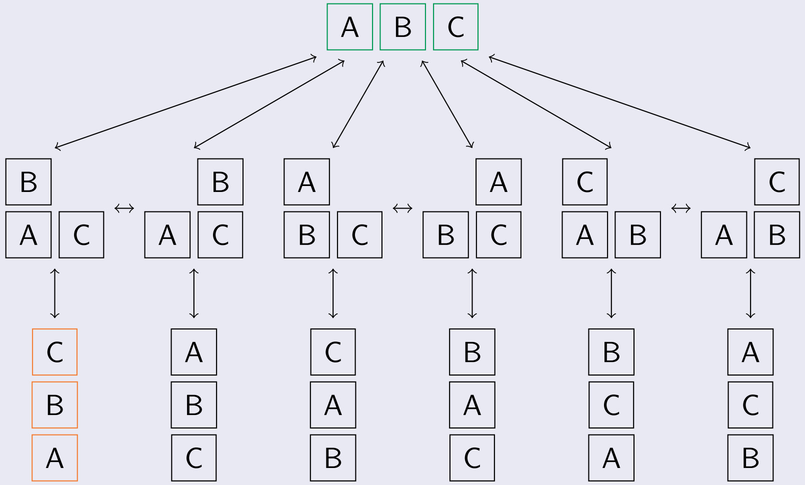
Sufficient Conditions

Bidirectional Search: ?



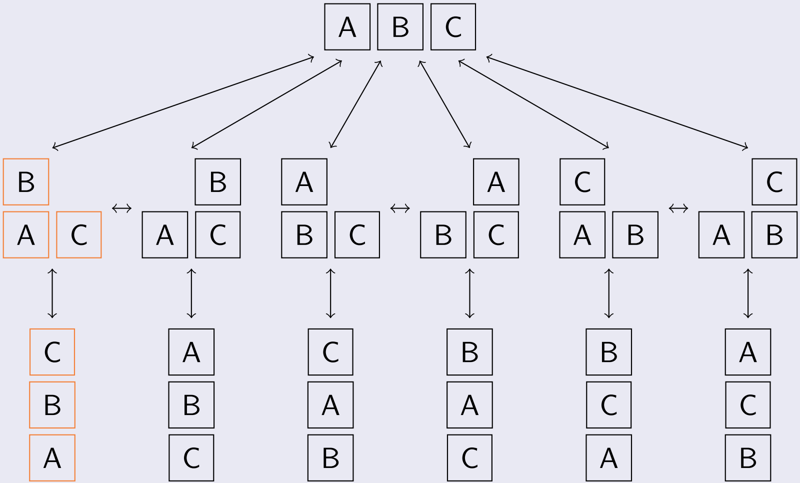
Sufficient Conditions

Bidirectional Search: ?



Sufficient Conditions

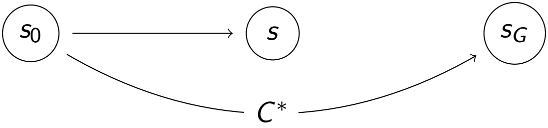
Bidirectional Search: ?



Sufficient Conditions

Unidirectional Search: Sufficient Condition

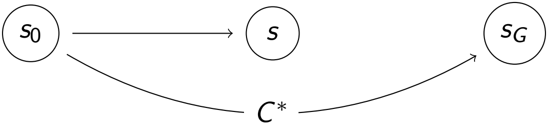
$$g(s) + h(s) < C^*$$



Sufficient Conditions

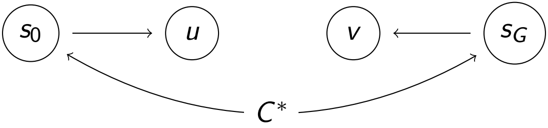
Unidirectional Search: Sufficient Condition

$$g(s) + h(s) < C^*$$



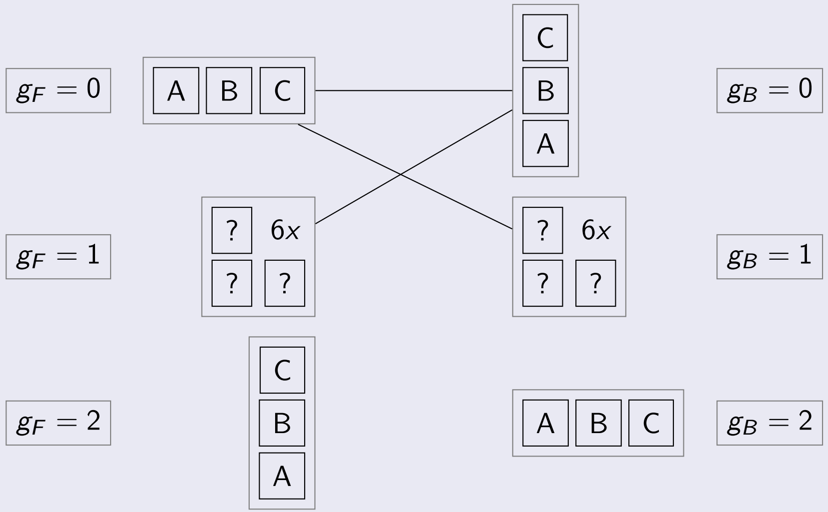
Bidirectional Search: Must-Expand Pairs

$$max\{f_F(u), f_B(v), g(u) + g(v)\} < C^*$$



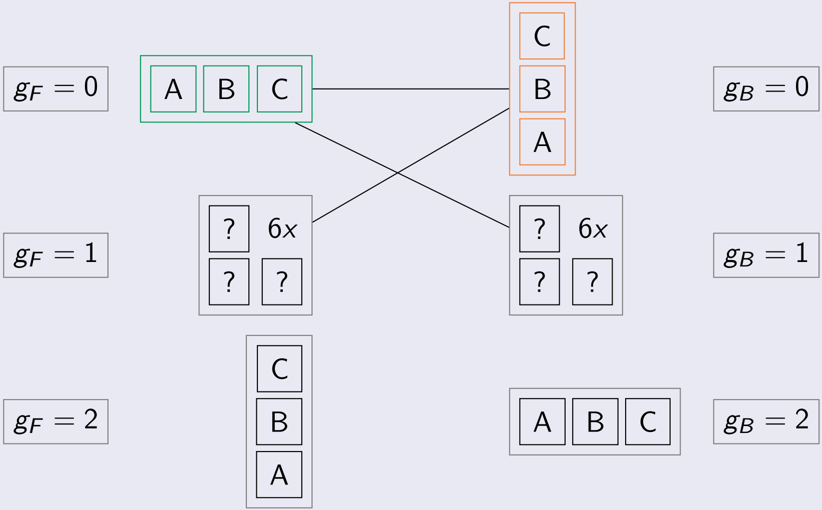
Sufficient Conditions

Must-Expand Graph: G_{MX}



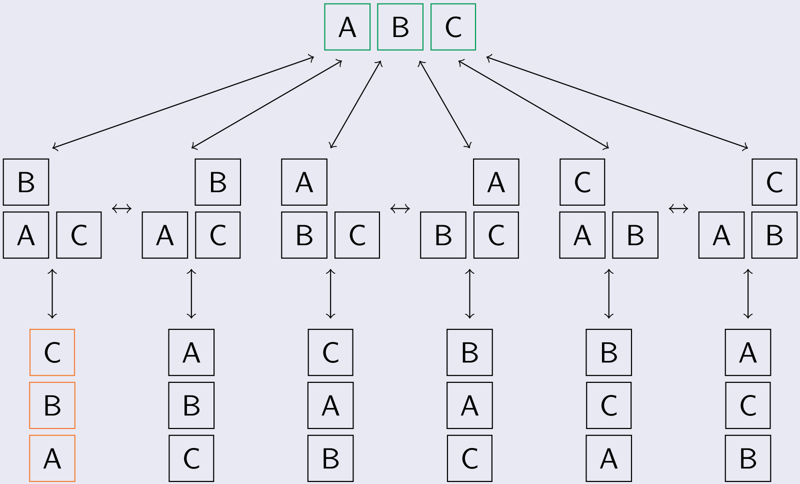
Sufficient Conditions

Minimal Vertex Cover of G_{MX} (VC)



Sufficient Conditions

Bidirectional Search: Optimal Expansion



NBS

Algorithm 1 NBS

```
Put s_0 in Open_F and s_G in Open_B
while Open_F and Open_B are not empty do
    Among u ∈ Open_F and v ∈ Open_B
    Select the pair (u, v) with lowest lb(u, v)
    if lb(u, v) ≥ cost(U) then
        return U
    Expand both u and v
    if new path from s_0 to s_G is found then
        Update U if new path is better than previous
```

NBS

Key Properties

- Admissible DXBB Algorithm
- Optimal Worst-Case
- Near-Optimal General Case

Key Difference to Search

Backward search is not a mirrored forward search.

- Actions are not trivially reversible.
- There are *Secondary Initial States*.

NBS

Key Properties

- Admissible DXBB Algorithm
- Optimal Worst-Case
- Near-Optimal General Case

Key Difference to Search

Backward search is not a mirrored forward search.

- Actions are not trivially reversible.
- There are *Secondary Initial States*.

Outline

NBS Experiment

NBS Experiment

Goal

Evaluate performance of NBS compared to A^*.

Parameters

- Latest IPC Benchmark
- Implementation in Fast Downward
- Time limit: 30 minutes. Memory Limit: 3.5 GB.
- Heuristics
  - Blind Heuristic
  - Max Heuristic
  - Critical Path Heuristic ($m = 2$)
  - Landmark-Cut Heuristic

NBS Experiment

Goal

Evaluate performance of NBS compared to A^*.

Parameters

- Latest IPC Benchmark
- Implementation in Fast Downward
- Time limit: 30 minutes. Memory Limit: 3.5 GB.
- Heuristics
  - Blind Heuristic
  - Max Heuristic
  - Critical Path Heuristic ($m = 2$)
  - Landmark-Cut Heuristic

Results and Evaluation

NBS > A^*

Domain	Algo	Solved Problems (#) h^1	h^{max}	h^m	h^{lmcut}
Blocks	A^*	18	21	10	28
	NBS	**27**	**27**	**13**	**31**

Similar Domains (~ 10)

- Driverlog
- Floortile-opt11-strips
- ..

Results and Evaluation

NBS > A^*

Domain	Algo	Solved Problems (#) h^1	h^{max}	h^m	h^{lmcut}
Blocks	A^*	18	21	10	28
	NBS	**27**	**27**	**13**	**31**

Similar Domains ($\sim$ 10)

- Driverlog
- Floortile-opt11-strips
- ..

Results and Evaluation

NBS $<$ A^*

Domain	Algo	Solved Problems (#)			
	Algo	h^1	h^{max}	h^m	h^{lmcut}
Storage	A^*	**14**	**15**	**7**	**15**
	NBS	5	5	2	4

Similar Domains ($\sim$ 20)

- Depot
- Elevators-opt11-strips
- ..

Results and Evaluation

NBS $< A^*$

Domain		Solved Problems (#)			
	Algo	h^1	h^{max}	h^m	h^{lmcut}
Storage	A^*	**14**	**15**	**7**	**15**
	NBS	5	5	2	4

Similar Domains ($\sim$ 20)

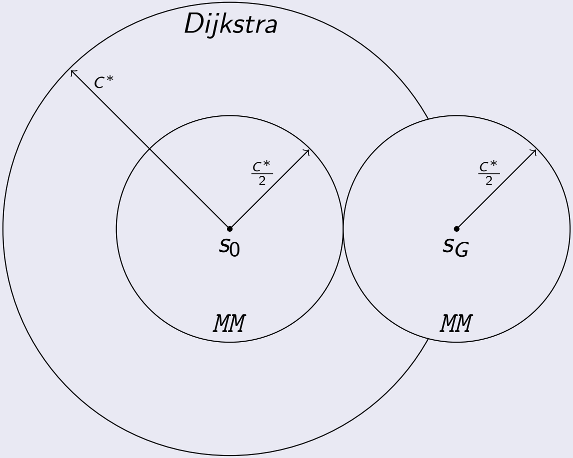
- Depot
- Elevators-opt11-strips
- ..

Outline

fMM Experiment

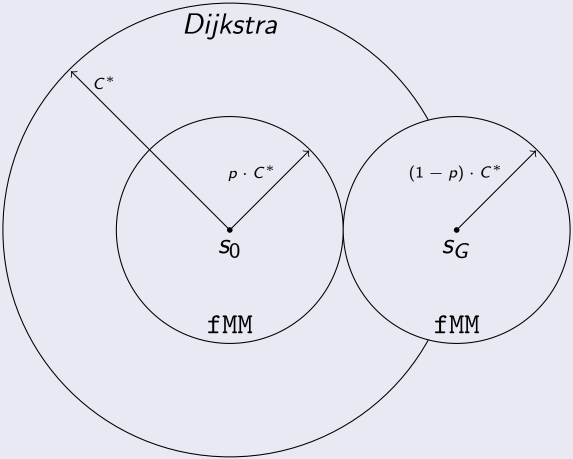
Meet-In-The-Middle

State Space



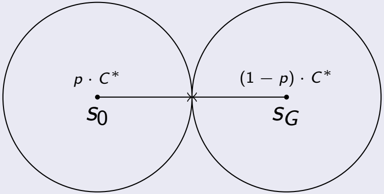
Fractional Meet-In-The-Middle

$p = 0.5$

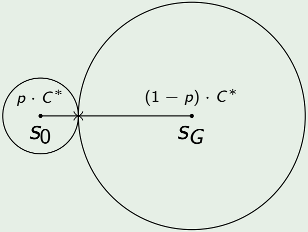


Fractional Meet-In-The-Middle

$p = 0.5$



$p = 0.25$



Fractional Meet-In-The-Middle

Key Properties

- Expands $|VC|$ states given input p^*.
- Reflects the structure of the state space.

Experiment Goal

Inquire the relation between the state space structure and the performance of NBS.

Experiment Parameters

- Same constraints as the NBS experiment.
- Equidistant p-value from 0 to 1 with step size 0.1.

Fractional Meet-In-The-Middle

Key Properties

- Expands $|VC|$ states given input p^*.
- Reflects the structure of the state space.

Experiment Goal

Inquire the relation between the state space structure and the performance of NBS.

Experiment Parameters

- Same constraints as the NBS experiment.
- Equidistant p-value from 0 to 1 with step size 0.1.

Fractional Meet-In-The-Middle

Key Properties

- Expands $|VC|$ states given input p^*.
- Reflects the structure of the state space.

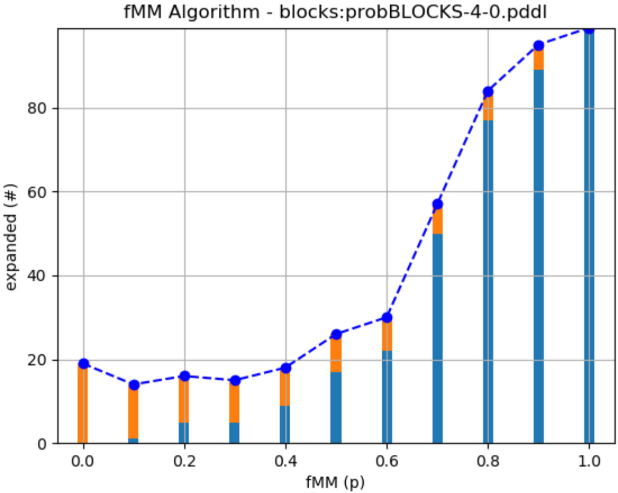
Experiment Goal

Inquire the relation between the state space structure and the performance of NBS.

Experiment Parameters

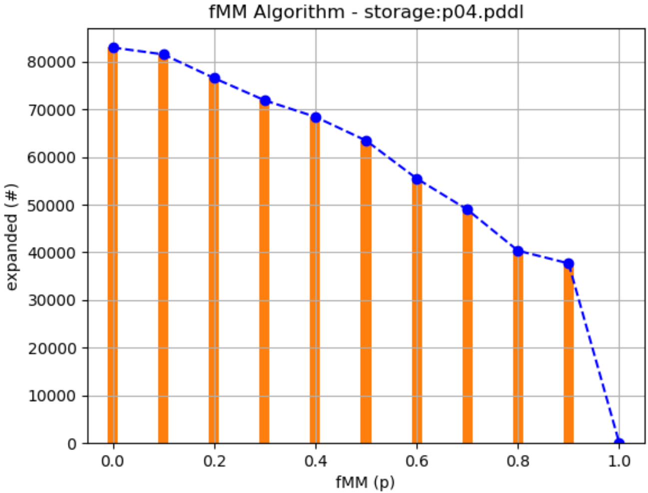
- Same constraints as the NBS experiment.
- Equidistant p-value from 0 to 1 with step size 0.1.

Results and Evaluation



Forward
Backward

Results and Evaluation



Forward
Backward

Results and Evaluation

Overview

- p^* is informative of the overall structure of the state space.
- The state space influences the performance of NBS.
- The structure of the state space correlates with the domain.

How can we use that information?

Case Study: Blocks Domain

Results and Evaluation

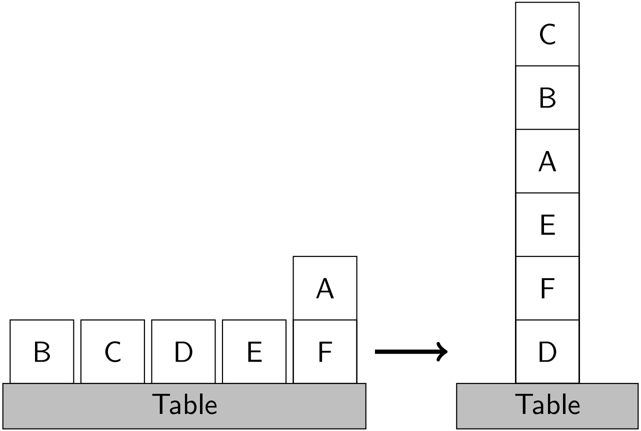
Overview

- p^* is informative of the overall structure of the state space.
- The state space influences the performance of NBS.
- The structure of the state space correlates with the domain.

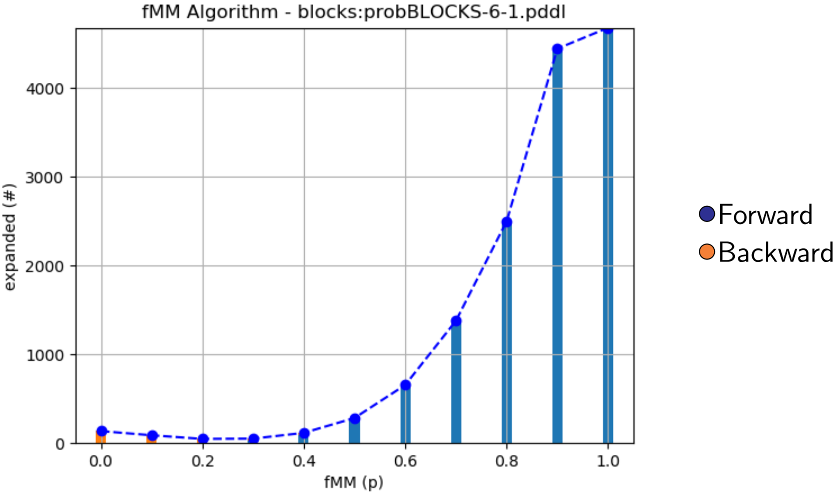
How can we use that information?

Case Study: Blocks Domain

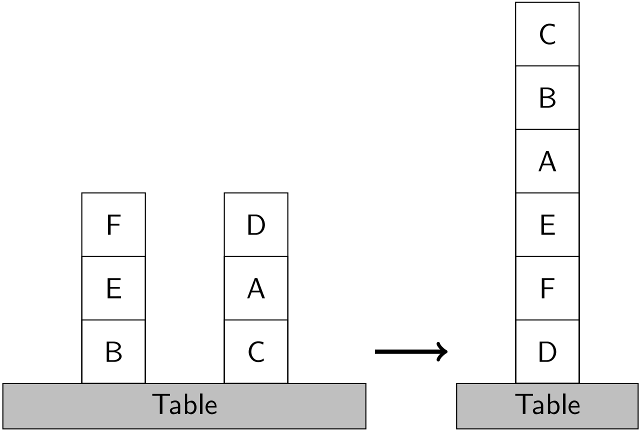
Case Study: Blocks Instance 6-1



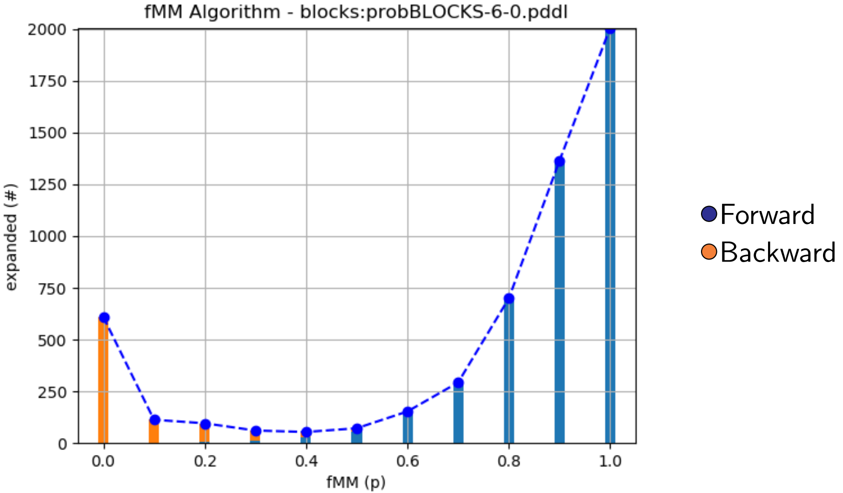
Case Study: Blocks Instance 6-1



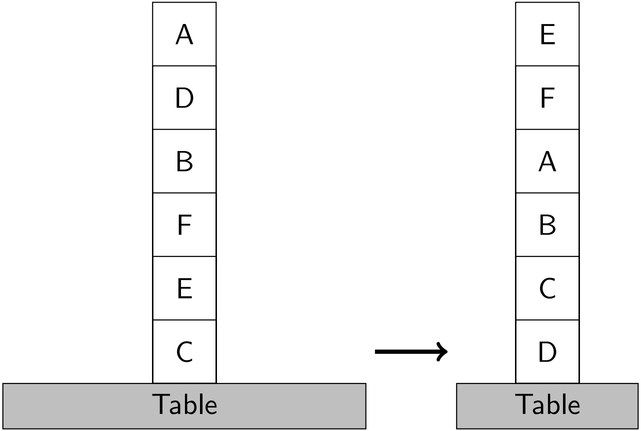
Case Study: Blocks Instance 6-0



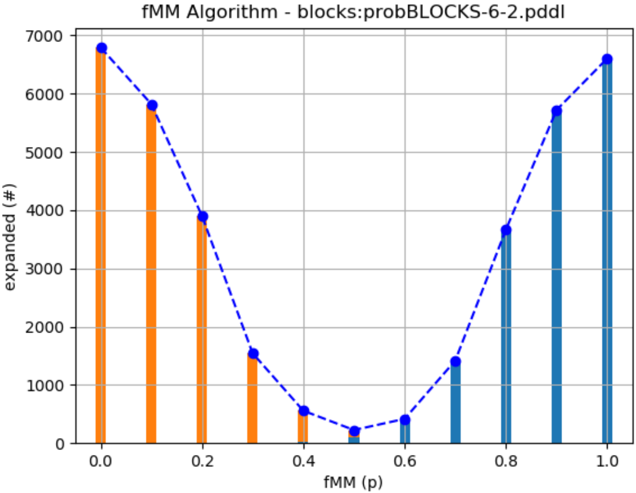
Case Study: Blocks Instance 6-0



Case Study: Blocks Instance 6-2



Case Study: Blocks Instance 6-2



Forward
Backward

Conclusion

Take Home Message

- Using NBS in planning can compete with A^*.
- The efficiency of NBS depends on the structure of the transition system.
- The predisposition towards certain search approaches correlates with the domain.

Conclusion

Take Home Message

- Using NBS in planning can compete with A^*.
- The efficiency of NBS depends on the structure of the transition system.
- The predisposition towards certain search approaches correlates with the domain.

Conclusion

Take Home Message

- Using NBS in planning can compete with A^*.
- The efficiency of NBS depends on the structure of the transition system.
- The predisposition towards certain search approaches correlates with the domain.

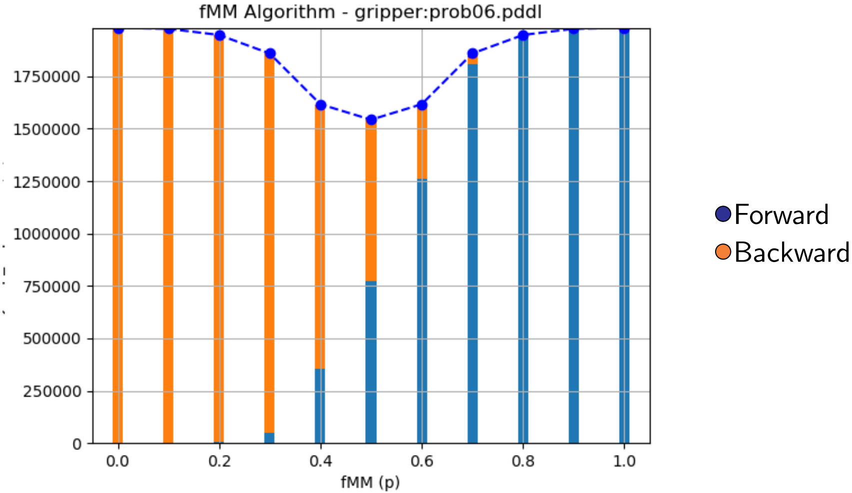
Questions?

For Further Reading I

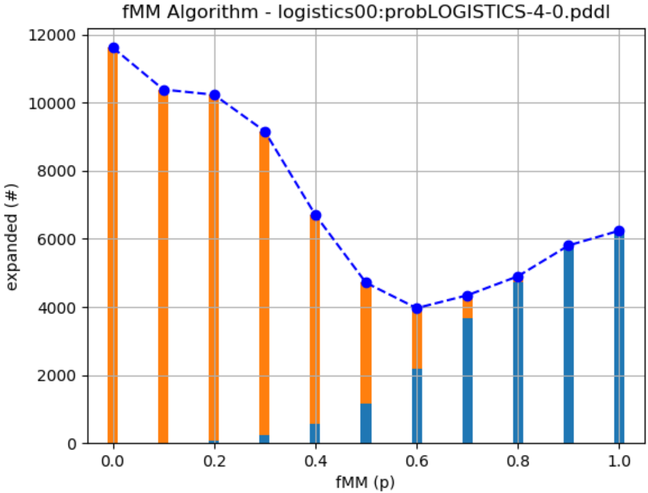
J. Chen, R. C. Holte, S. Zilles, N. Sturtevant
Front-to-end bidirectional heuristic search with near-optimal node expansions.
IJCAI 2017, pages: 489-495, AAAI Press.

J. Eckerle, J. Chen, R. C. Holte, S. Zilles
Sufficient conditions for node expansion in bidirectional heuristic search.
ICAPS 2017, pages: 79-87, AAAI Press.

Gripper Instance 06

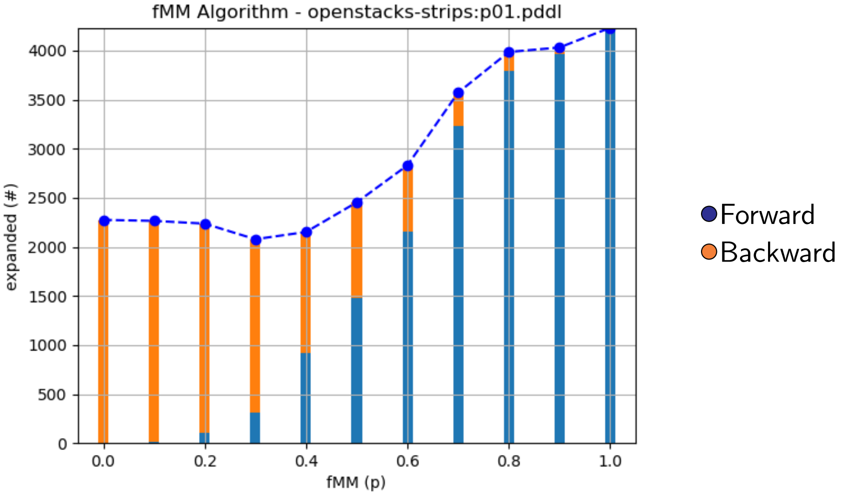


Logistics00 Instance 4-0

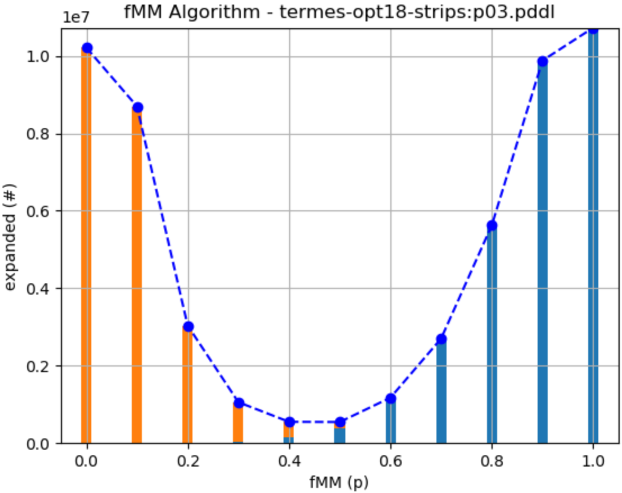


Forward
Backward

Openstacks-Strips Instance 01



Termes Instance 03



Forward
Backward