Precomputing Multi-Agent Path Replanning Using Temporal Flexibility

Introducing FlexSIPP

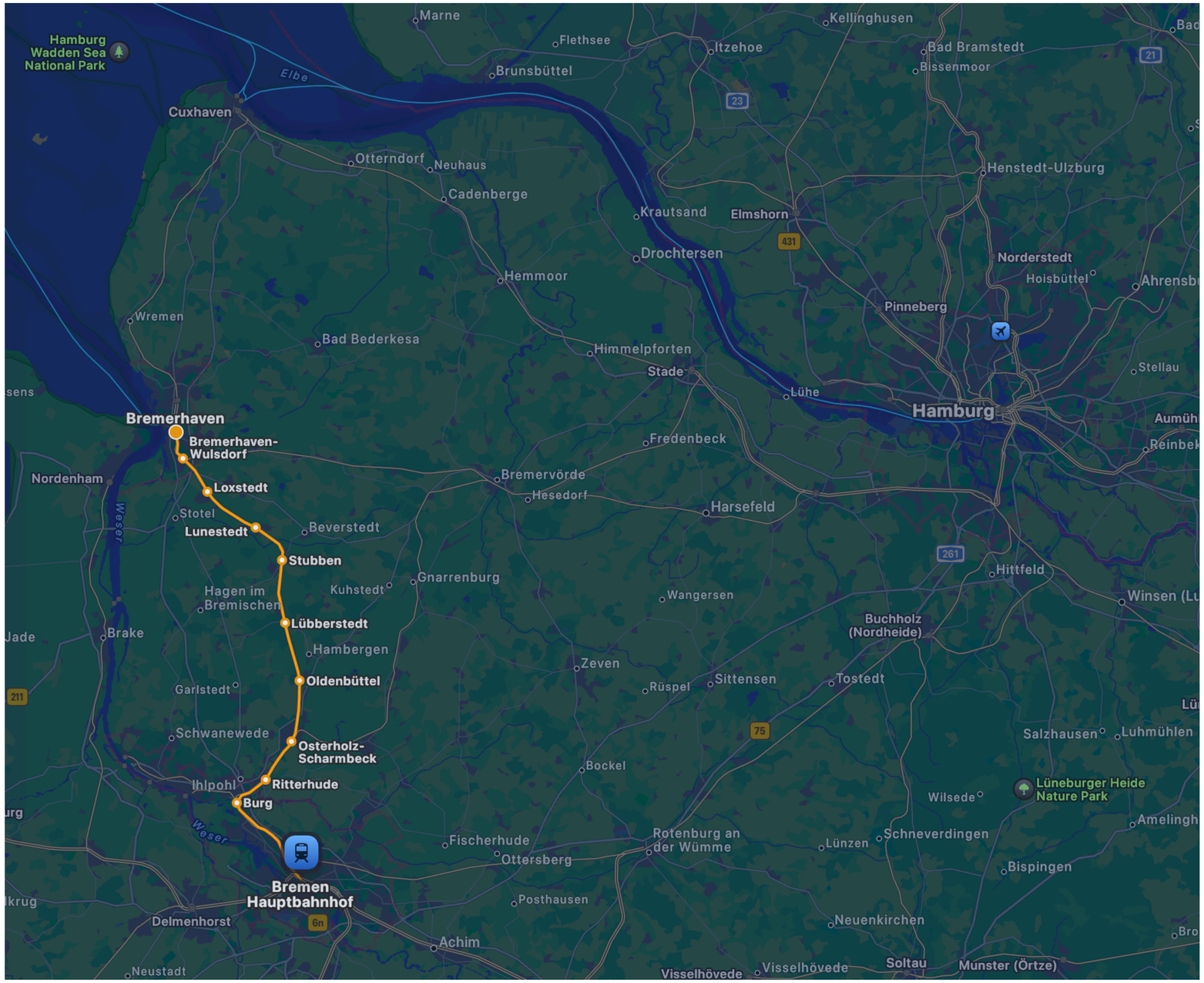
Issa Hanou, Eric Kemmeren, Devin Thomas and Mathijs de Weerdt

August 16, 2026

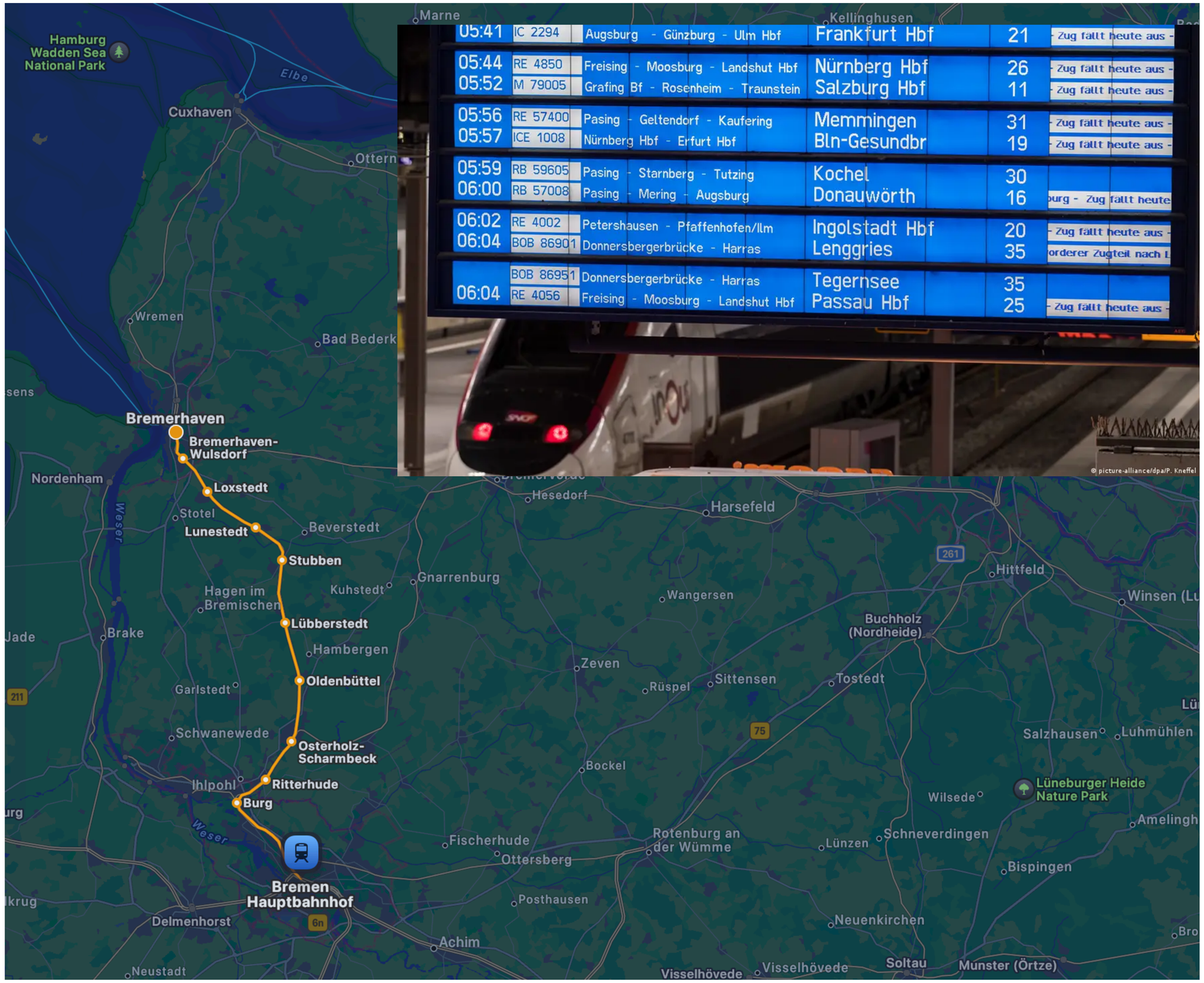
SoCS'26

ProRail University of New Hampshire TU Delft

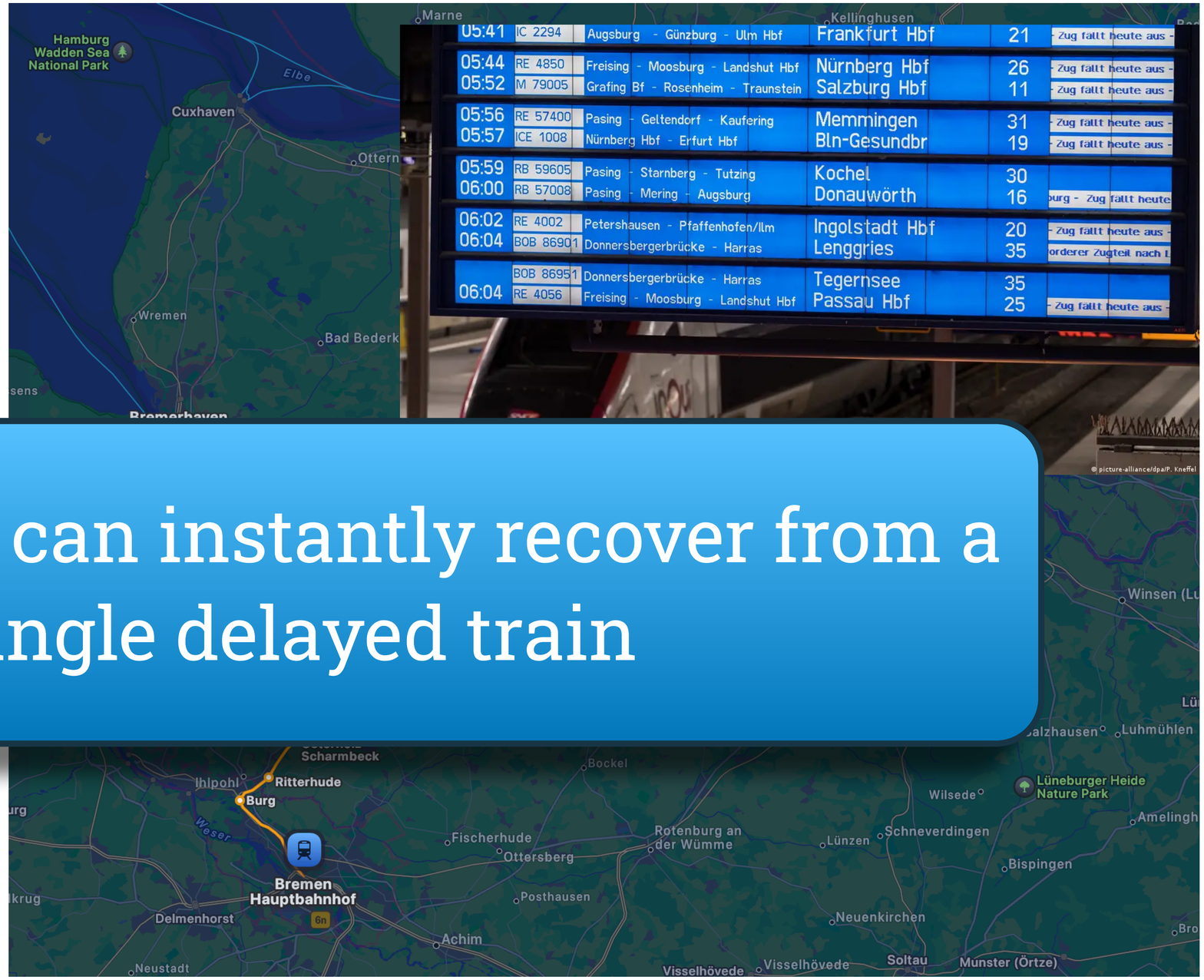
Motivation


Hamburg Wadden Sea National Park
Elbe
Cuxhaven
Marne
Flethsee
Brunsbüttel
Itzehoe
Kellinghusen
Bad Bramstedt
Bissenmoor
Henstedt-Ulzburg
Otterndorf
Neuhaus
Cadenberge
Krautsand
Elmshorn
Drochtersen
Norderstedt
Hoisbüttel
Ahrensb
Hemmoor
Pinneberg
Wremen
Bad Bederkesa
Himmelpforten
Stade
Stellau
Lühe
Hamburg
Bremerhaven
Bremerhaven-Wulsdorf
Loxstedt
Fredenbeck
Nordenham
Bremervörde
Hesedorf
Harsefeld
Weser
Stotel
Lunestedt
Beverstedt
Stubben
Hittfeld
Hagen im Bremischen
Kuhstedt
Gnarrenburg
Wangersen
Winsen (Lu
Lübberstedt
Brake
Buchholz (Nordheide)
Hambergen
Zeven
Oldenbüttel
Rüspel
Sittensen
Tostedt
Garlstedt
Schwanewede
Osterholz-Scharmbeck
Salzhausen
Luhmühlen
Bockel
Ihlpohl
Ritterhude
Burg
Lüneburger Heide Nature Park
Wilsede
Weser
Fischerhude
Rotenburg an der Wümme
Lünzen
Schneverdingen
Amelingh
Ottersberg
Bispingen
Bremen Hauptbahnhof
Posthausen
Delmenhorst
Neuenkirchen
Achim
Visselhövede
Soltau
Munster (Örtze)
Neustadt

Motivation


Hamburg Wadden Sea National Park
Elbe
Cuxhaven
Ottern
Marne
Kellinghusen
Wremen
Bad Bederk
Bremerhaven
Bremerhaven-Wulsdorf
Loxstedt
Nordenham
Stotel
Lunestedt
Beverstedt
Stubben
Hagen im Bremischen
Kuhstedt
Gnarrenburg
Lübberstedt
Brake
Jade
Hambergen
Oldenbüttel
Garlstedt
Schwanewede
Osterholz-Scharmbeck
Ihlpohl
Ritterhude
Burg
Weser
Bremen Hauptbahnhof
Delmenhorst
Neustadt
Achim
Posthausen
Fischerhude
Ottersberg
Rotenburg an der Wümme
Bockel
Zeven
Rüspel
Sittensen
Tostedt
Wangersen
Buchholz (Nordheide)
Hittfeld
Winsen (Lu
Harsefeld
Hesedorf
Salzhausen
Luhmühlen
Lüneburger Heide Nature Park
Wilsede
Amelingh
Bispingen
Schneverdingen
Lünzen
Neuenkirchen
Visselhövede
Soltau
Munster (Örtze)
261
211
75
6n
05:41 IC 2294 Augsburg - Günzburg - Ulm Hbf Frankfurt Hbf 21 Zug fällt heute aus -
05:44 RE 4850 Freising - Moosburg - Landshut Hbf Nürnberg Hbf 26 Zug fällt heute aus -
05:52 M 79005 Grafing Bf - Rosenheim - Traunstein Salzburg Hbf 11 Zug fällt heute aus -
05:56 RE 57400 Pasing - Geltendorf - Kaufering Memmingen 31 Zug fällt heute aus -
05:57 ICE 1008 Nürnberg Hbf - Erfurt Hbf Bln-Gesundbr 19 Zug fällt heute aus -
05:59 RB 59605 Pasing - Starnberg - Tutzing Kochel 30
06:00 RB 57008 Pasing - Mering - Augsburg Donauwörth 16 burg - Zug fällt heute
06:02 RE 4002 Petershausen - Pfaffenhofen/Ilm Ingolstadt Hbf 20 Zug fällt heute aus -
06:04 BOB 86901 Donnersbergerbrücke - Harras Lenggries 35 orderer Zugteil nach L
BOB 86951 Donnersbergerbrücke - Harras
06:04 RE 4056 Freising - Moosburg - Landshut Hbf Tegernsee 35
Passau Hbf 25 Zug fällt heute aus -
© picture-alliance/dpa/P. Kneffel

Motivation



Our method can instantly recover from a single delayed train

Overview

Overview

Recover instantly from single-agent delay in multi-agent execution

Overview

Recover instantly from single-agent delay in multi-agent execution

- Don’t affect other agents [Hanou & Wild, et al. 2024]

Overview

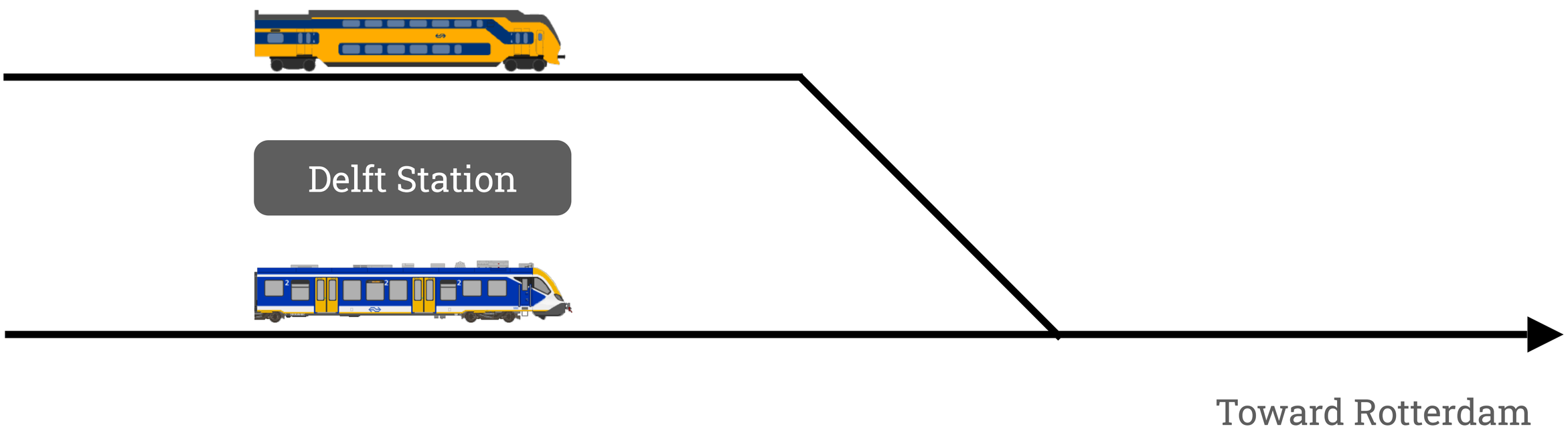
Recover instantly from single-agent delay in multi-agent execution

- Don’t affect other agents [Hanou & Wild, et al. 2024]

➢ Today: delay other agents, safely

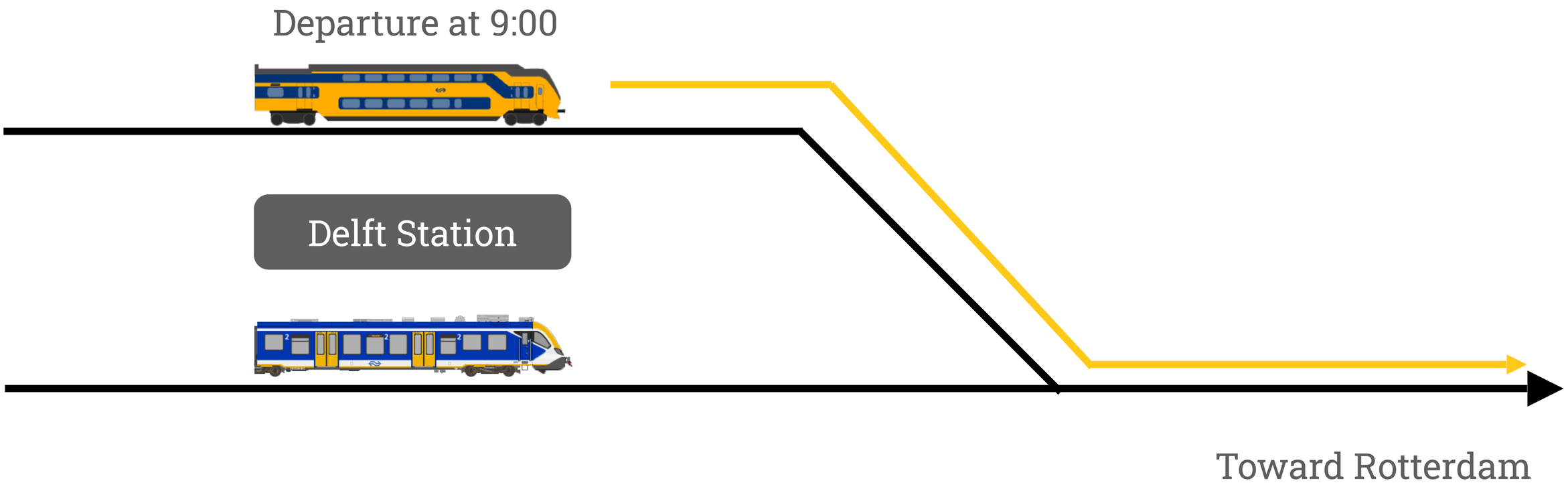
Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting



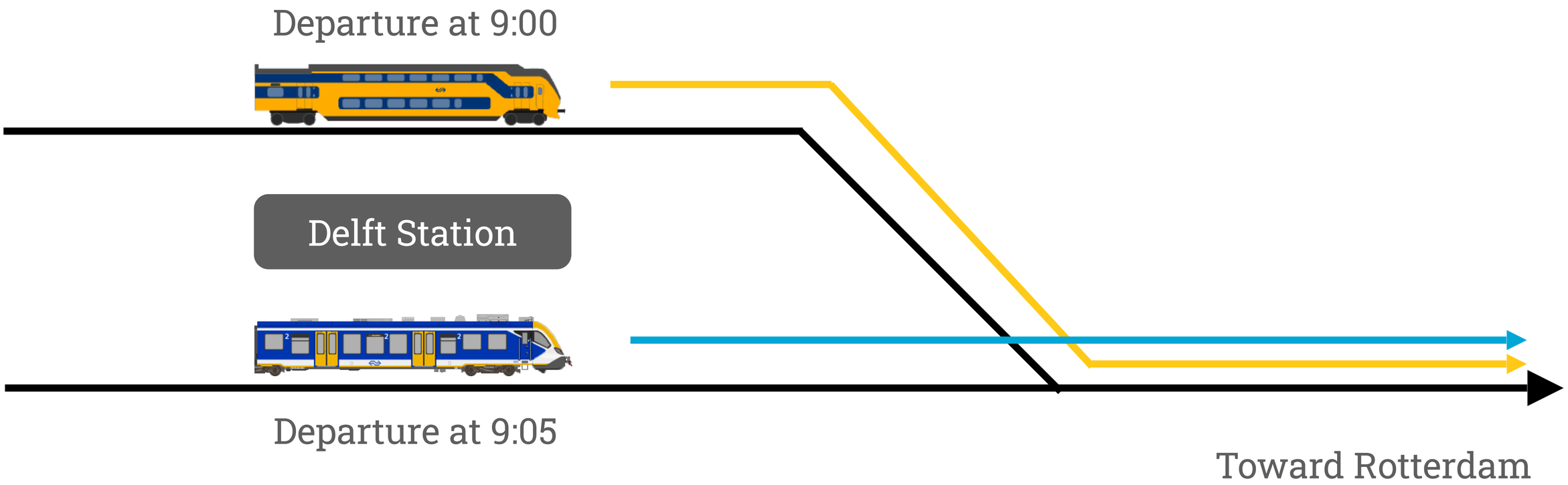
Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting



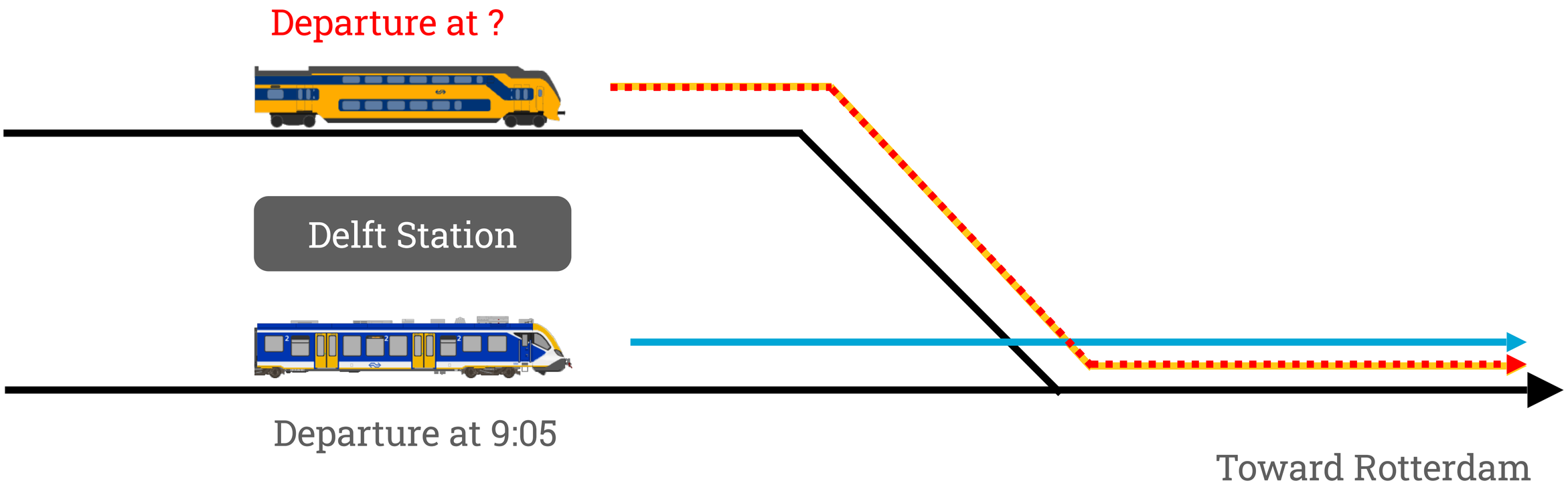
Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting



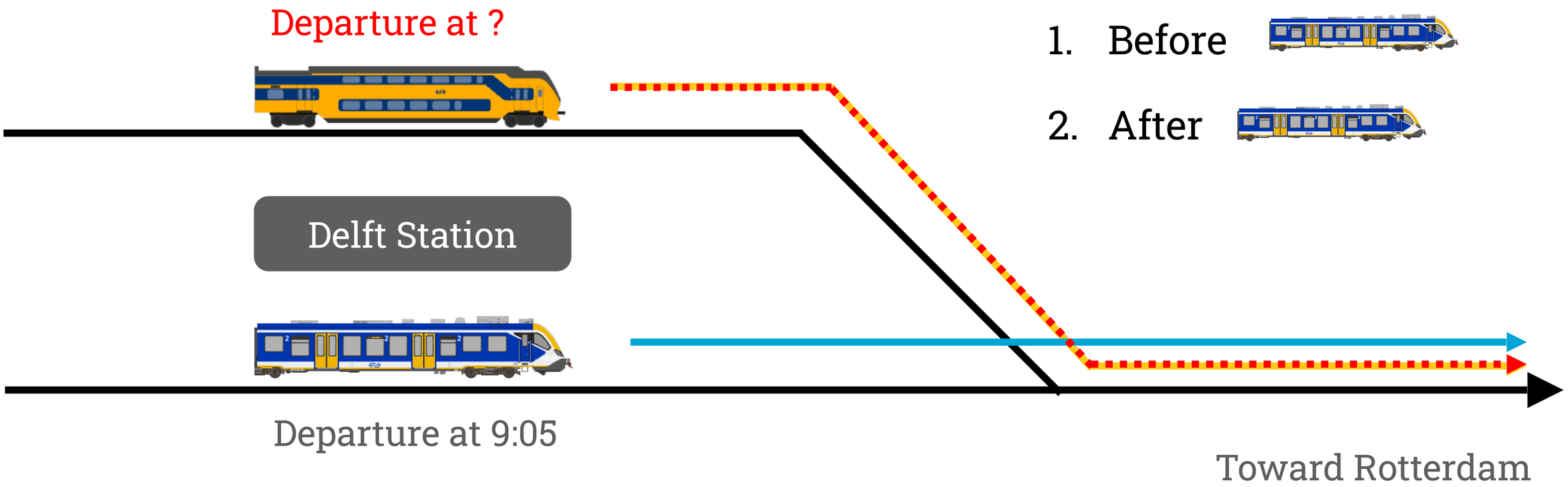
Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting



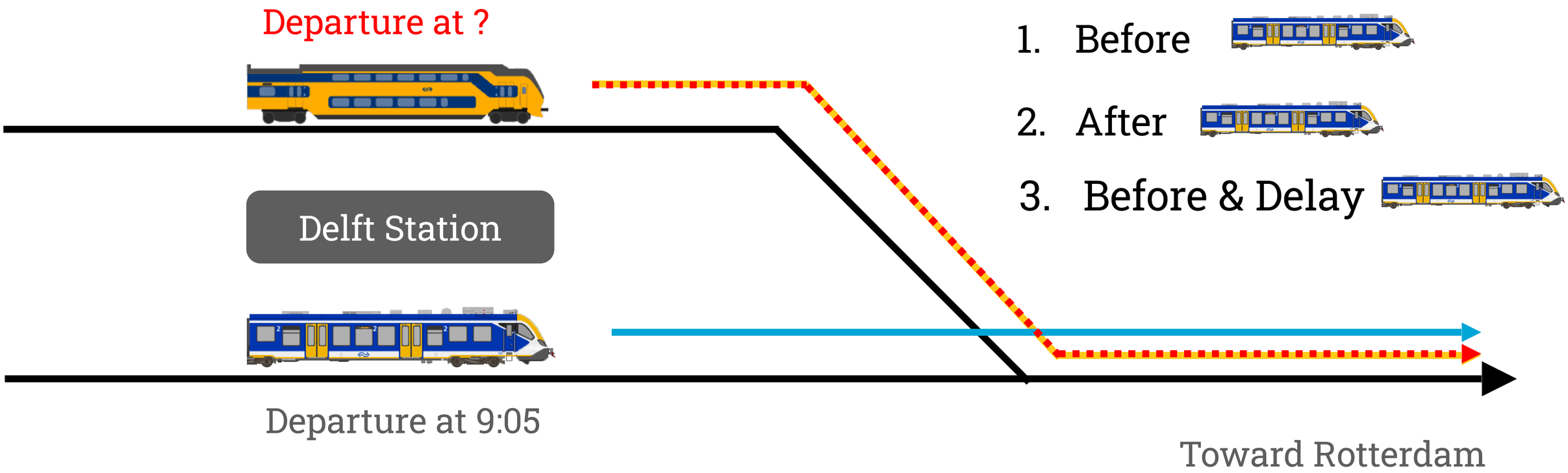
Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting



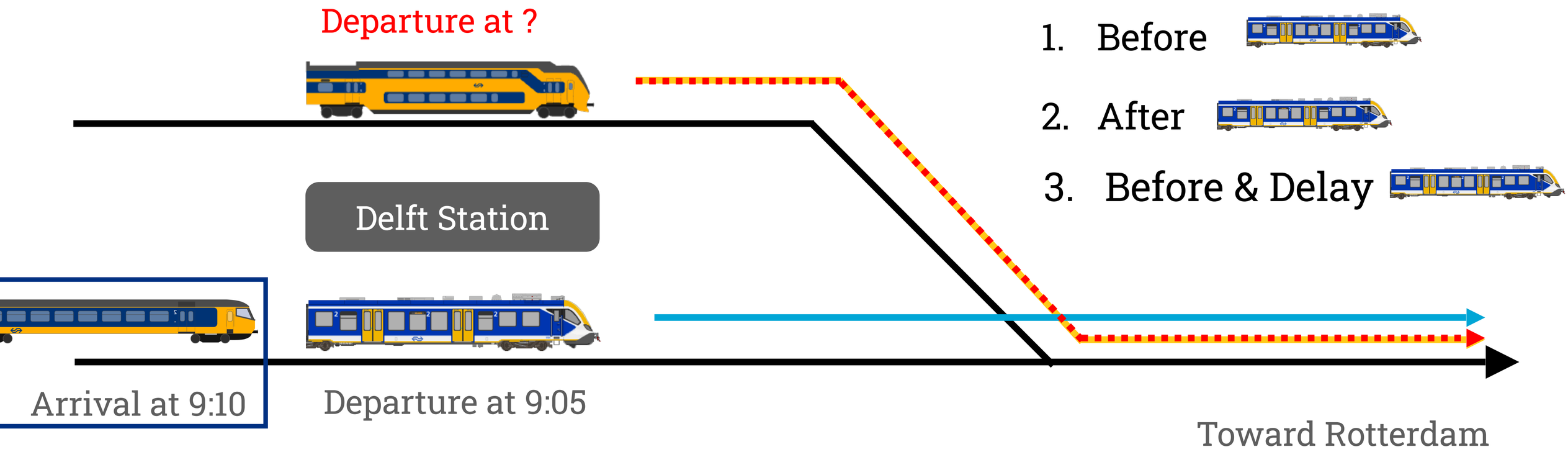
Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting



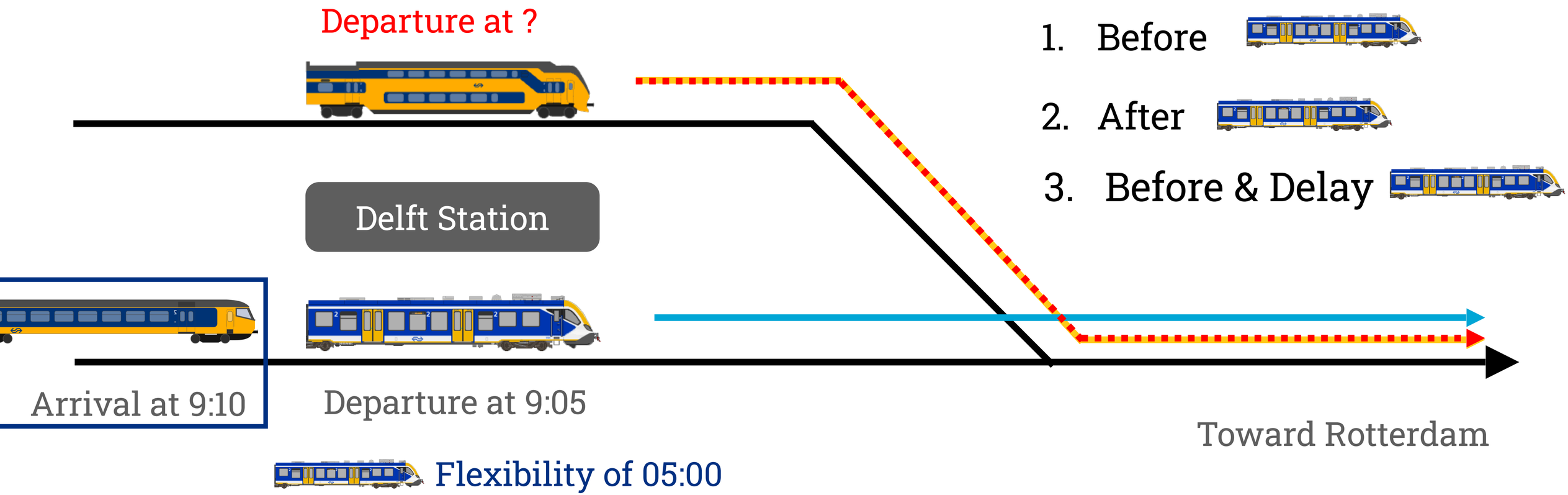
Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting



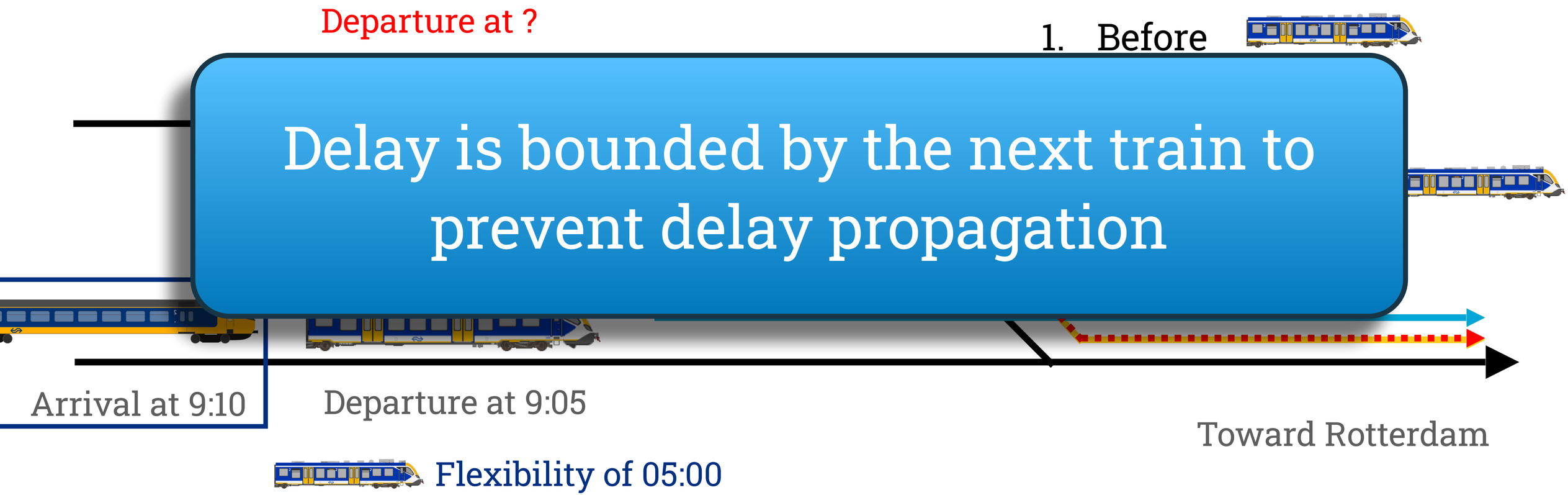
Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting

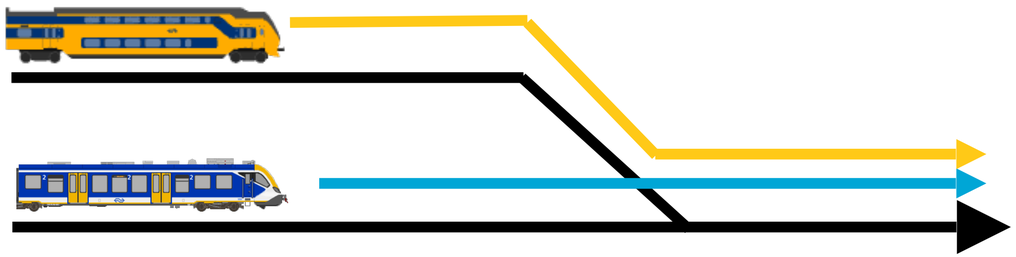


Multi-Agent Execution Delay Replanning (MAEDeR)

Single delayed agent in multi-agent setting

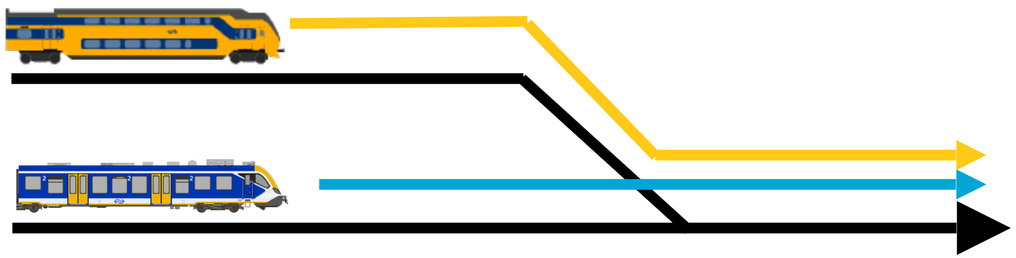


Replanning



Replanning

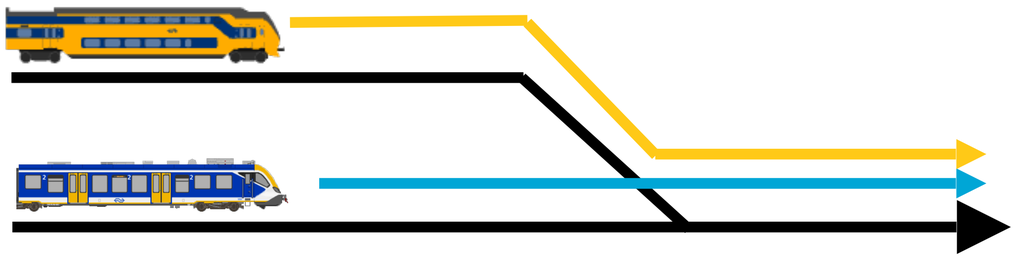
- Intuitively: replan on delay
- In execution: cannot pause world



Replanning

- Intuitively: replan on delay
- In execution: cannot pause world

➢(re)Planning takes time!



Replanning

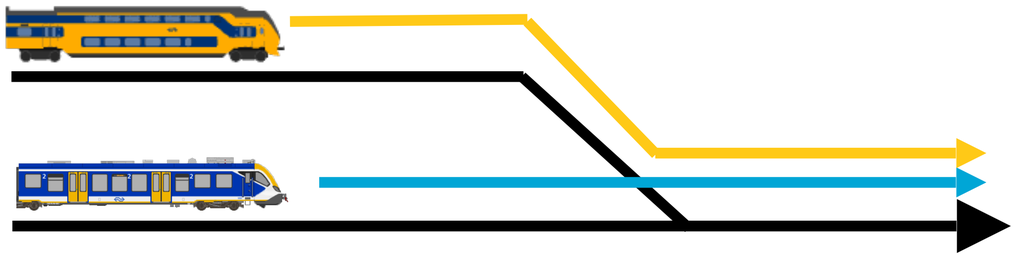
- Intuitively: replan on delay
- In execution: cannot pause world

  ➢(re)Planning takes time!

- Precompute any-start-time plan

  ➢Instantly recover new plan

@MAEDeR (Hanou & Thomas et al. 2024)



Replanning With Temporal Flexibility

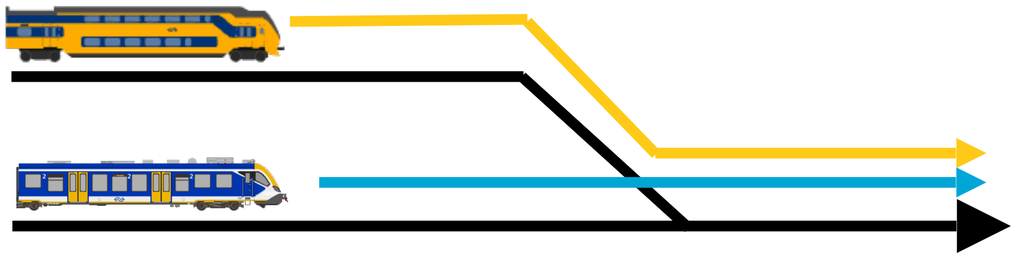
- Intuitively: replan on delay
- In execution: cannot pause world

  ➢(re)Planning takes time!

- Precompute any-start-time plan

  ➢Instantly recover new plan

@MAEDeR (Hanou & Thomas et al. 2024)



Replanning

- Intuitively: replan on delay
- In execution: cannot pause world

➢(re)Planning takes time!

With Temporal Flexibility

1. Precompute any-start-time plan

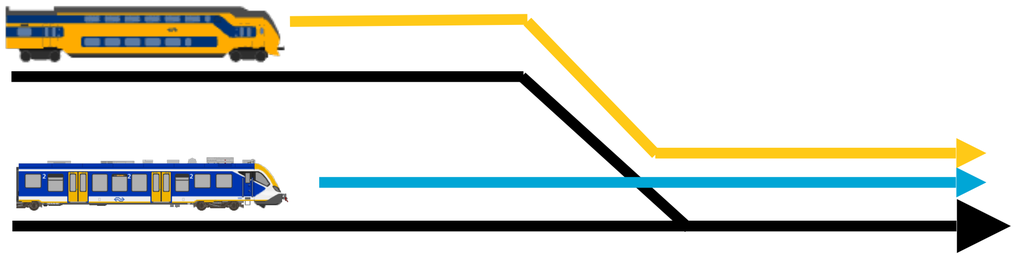
➢Instantly recover new plan

2. Query single optimal delay decision

- Precompute any-start-time plan

➢Instantly recover new plan

@MAEDeR (Hanou & Thomas et al. 2024)



Replanning

- Intuitively: replan on delay
- In execution: cannot pause world

➢(re)Planning takes time!

- Precompute any-start-time plan

➢Instantly recover new plan

@MAEDeR (Hanou & Thomas et al. 2024)

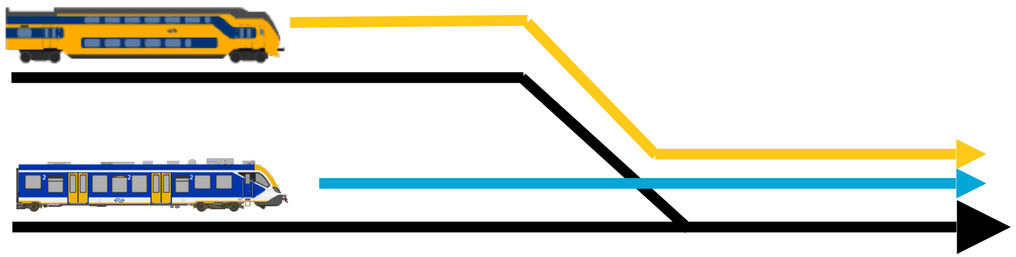
With Temporal Flexibility

1. Precompute any-start-time plan

➢Instantly recover new plan

2. Query single optimal delay decision

FlexSIPP: Each plan gives delay for other agents

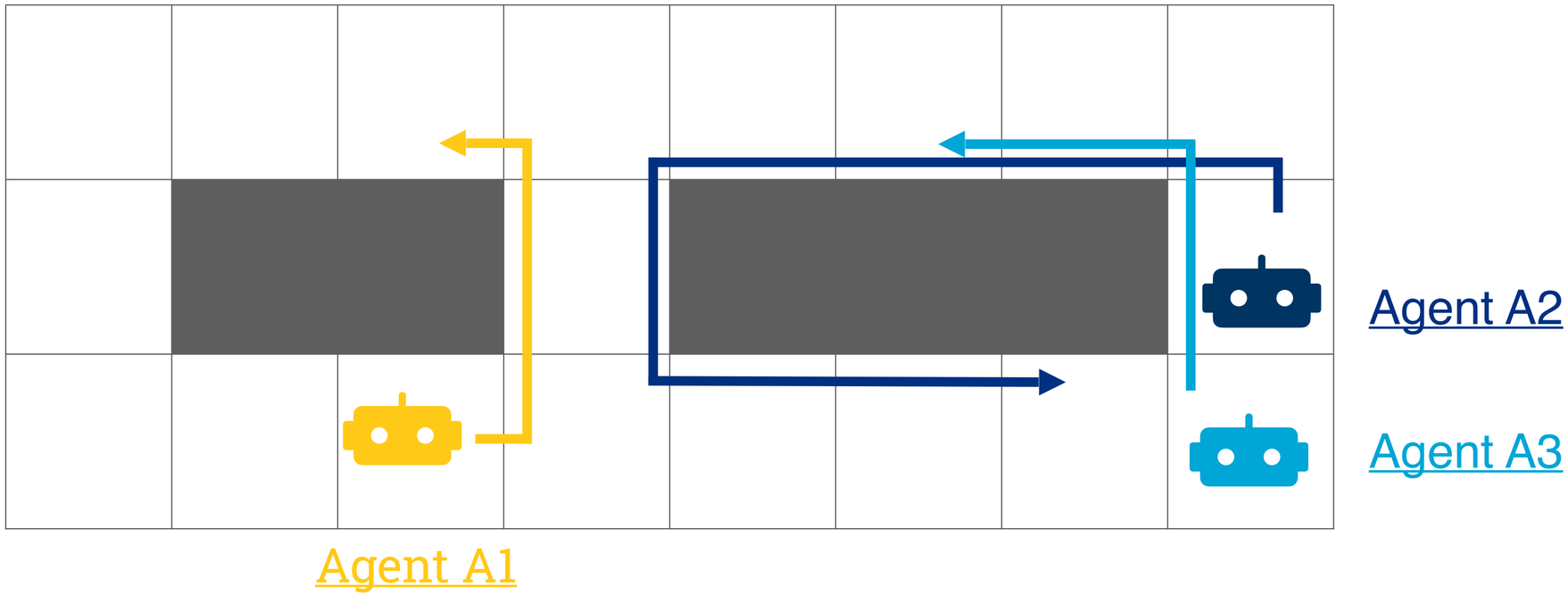


Problem features

- Safe Interval Path Planning (SIPP)
  - Continuous time and space in safe states and actions
- Bound the delay
  - Inherent in original plan
  - Prevent delay propagation

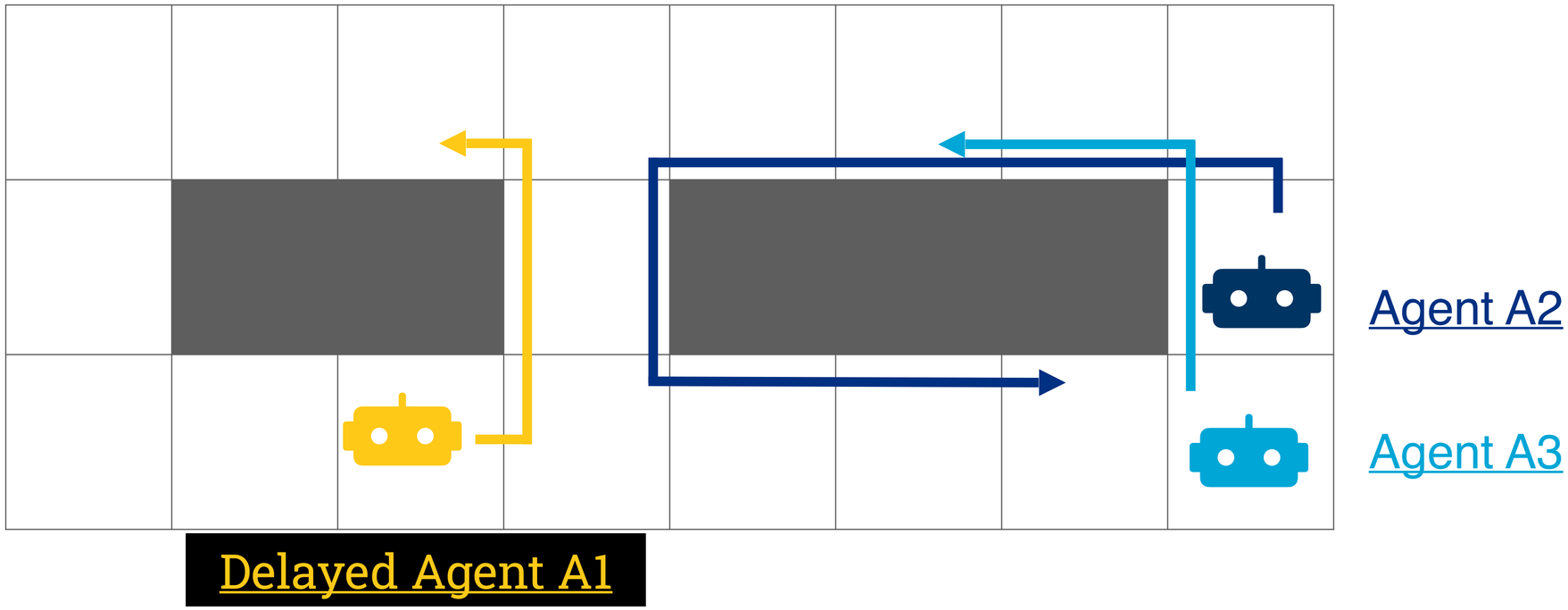
Multi-Agent Pathfinding

Constructing the SIPP graph



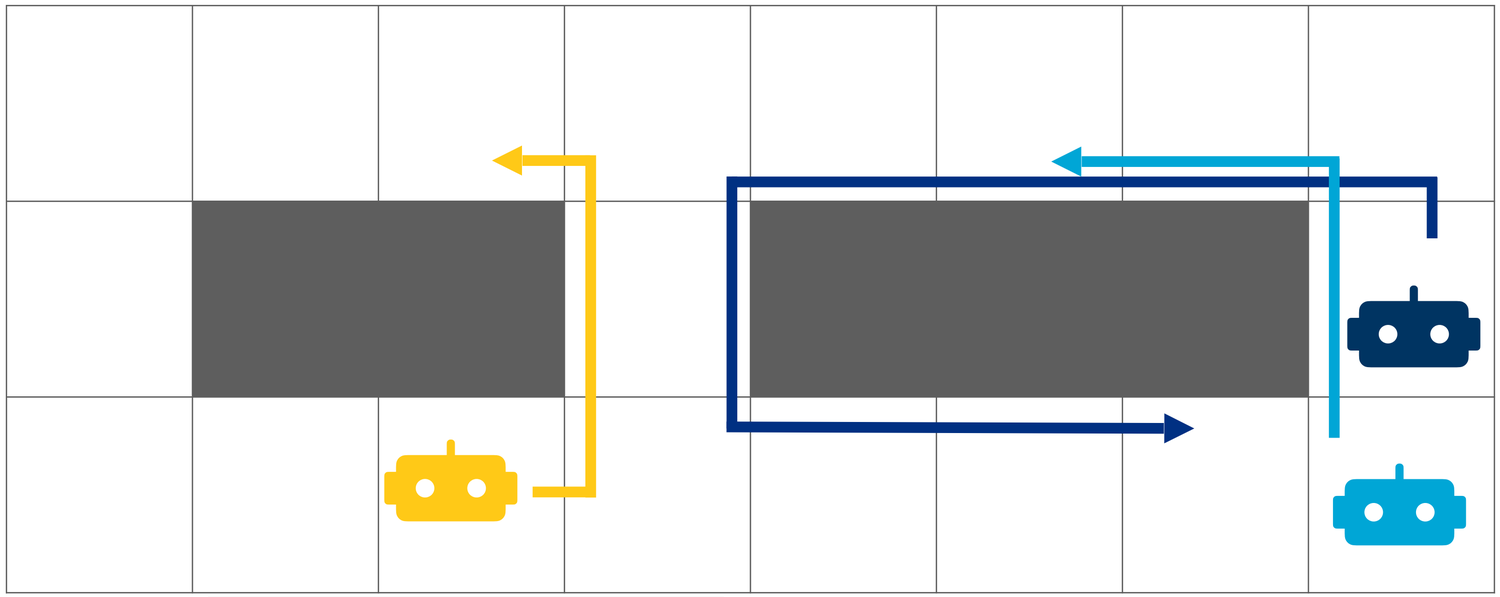
Multi-Agent Pathfinding

Constructing the SIPP graph



Multi-Agent Pathfinding

Constructing the SIPP graph



Agent A2

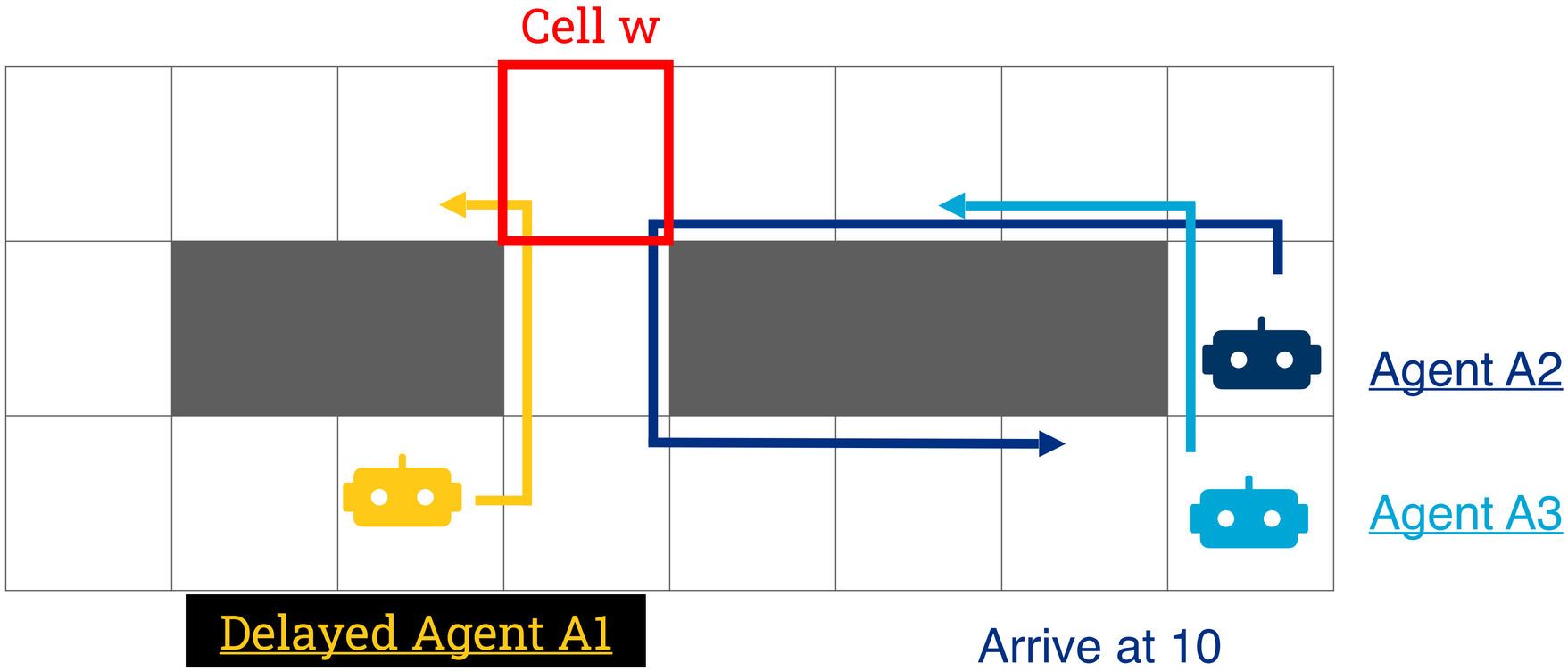
Agent A3

Delayed Agent A1

Arrive at 10
Deadline at 12

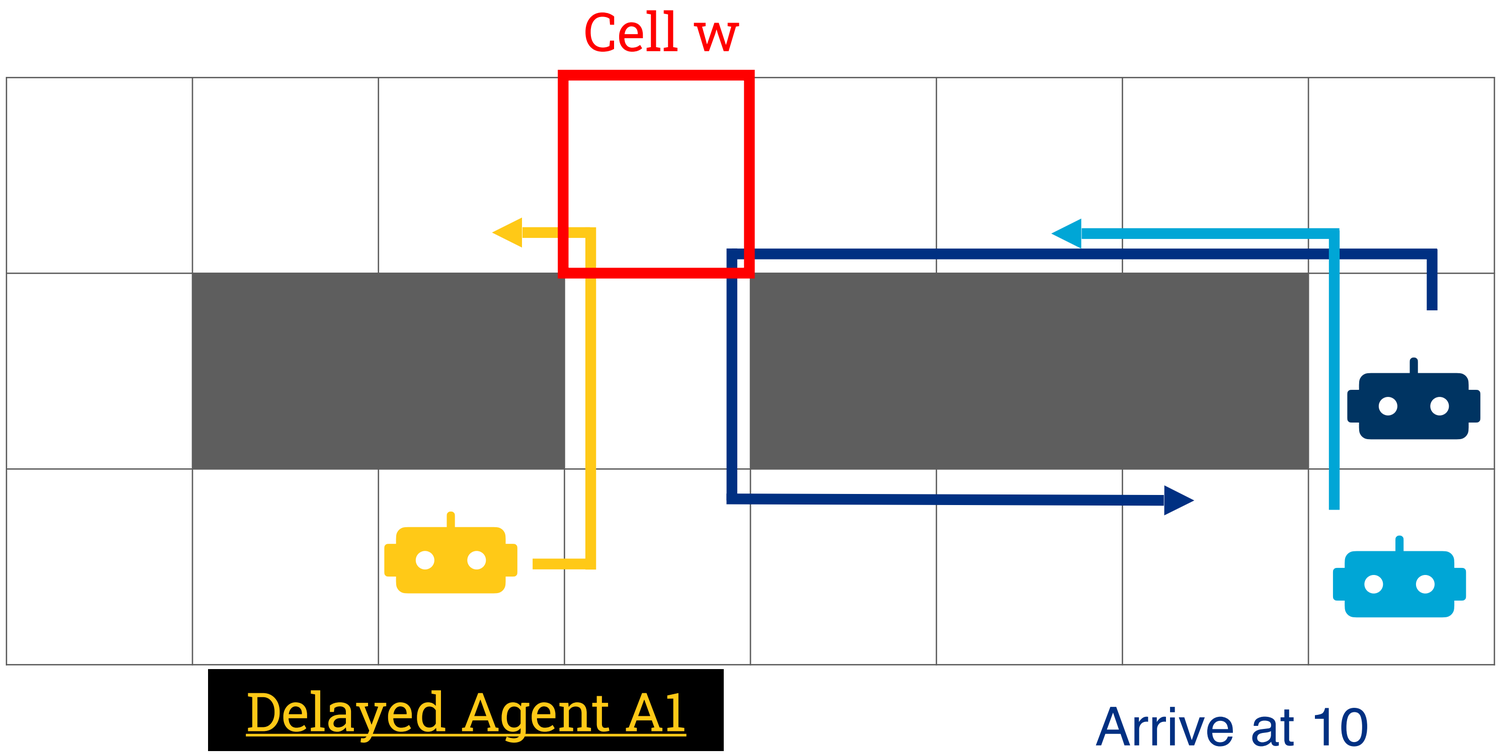
Multi-Agent Pathfinding

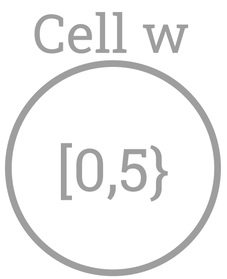
Constructing the SIPP graph



Multi-Agent Pathfinding

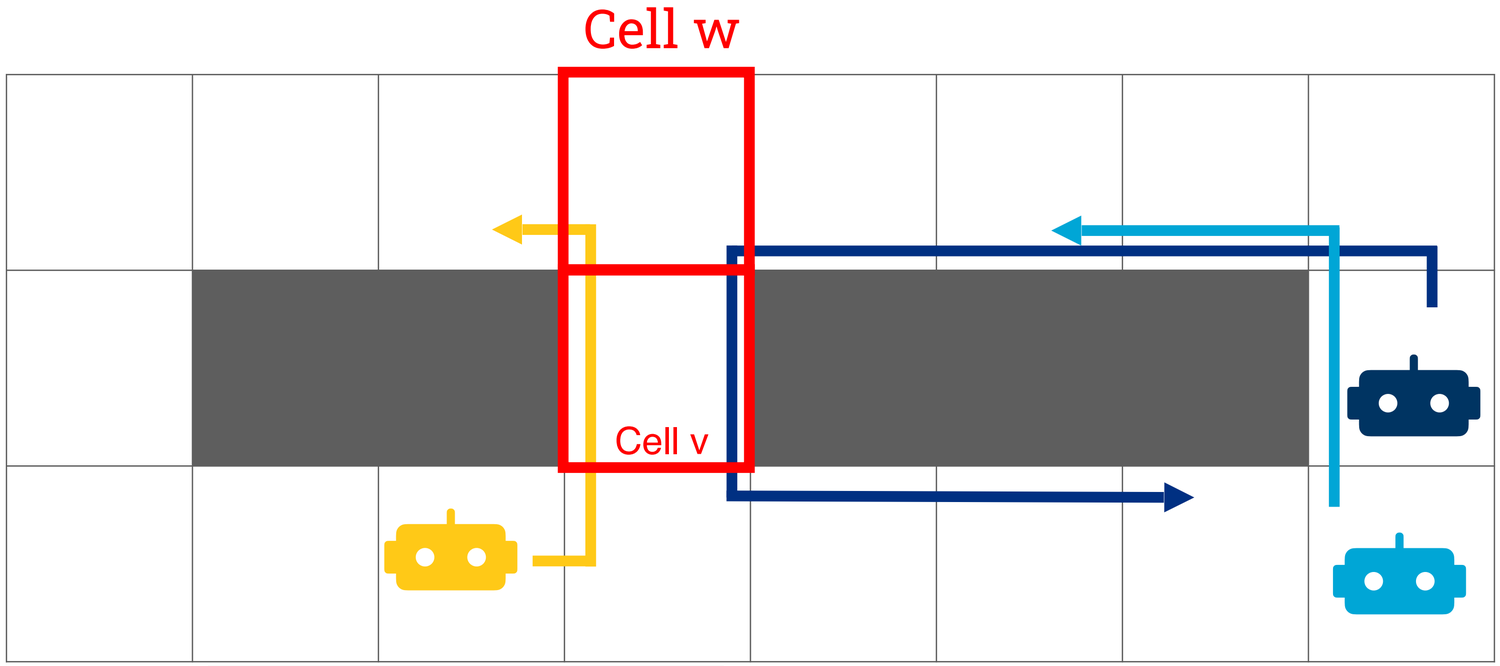
Constructing the SIPP graph

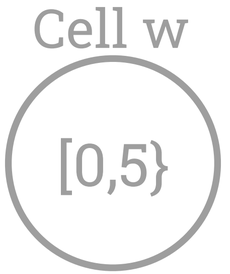




Multi-Agent Pathfinding

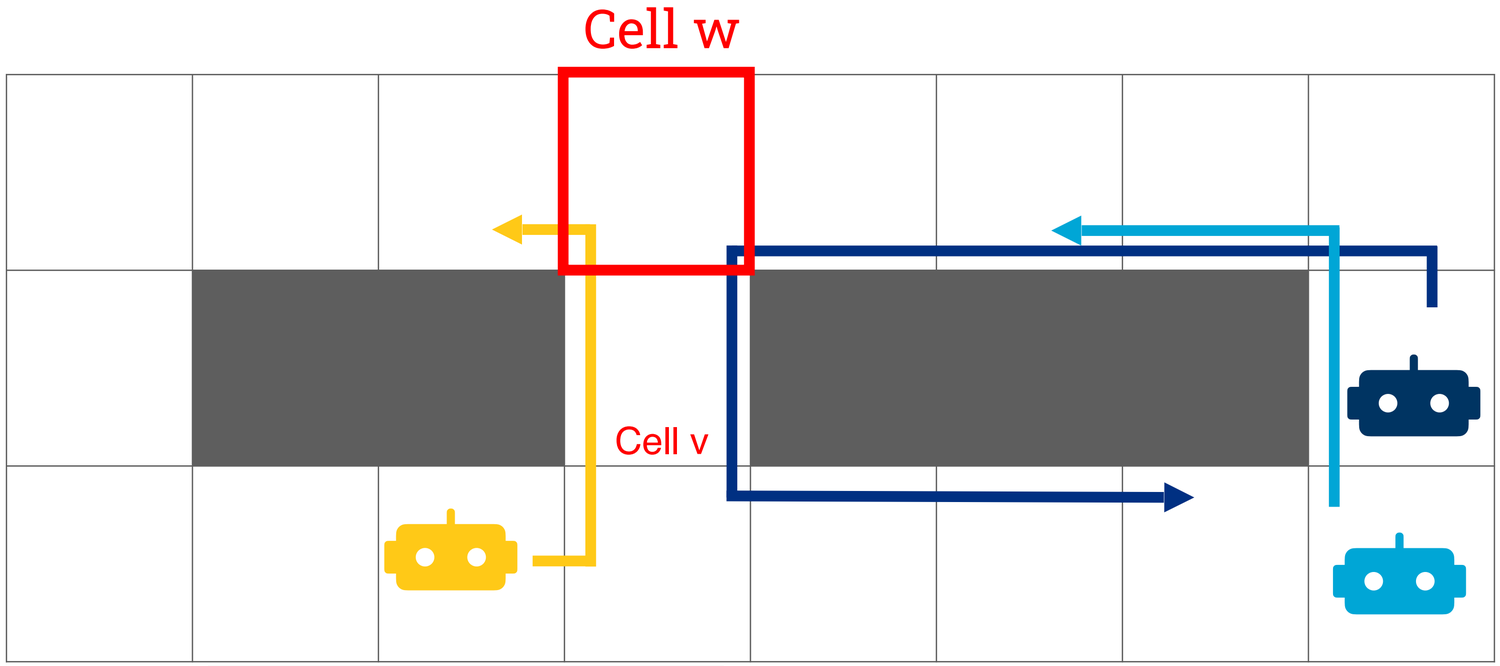
Constructing the SIPP graph

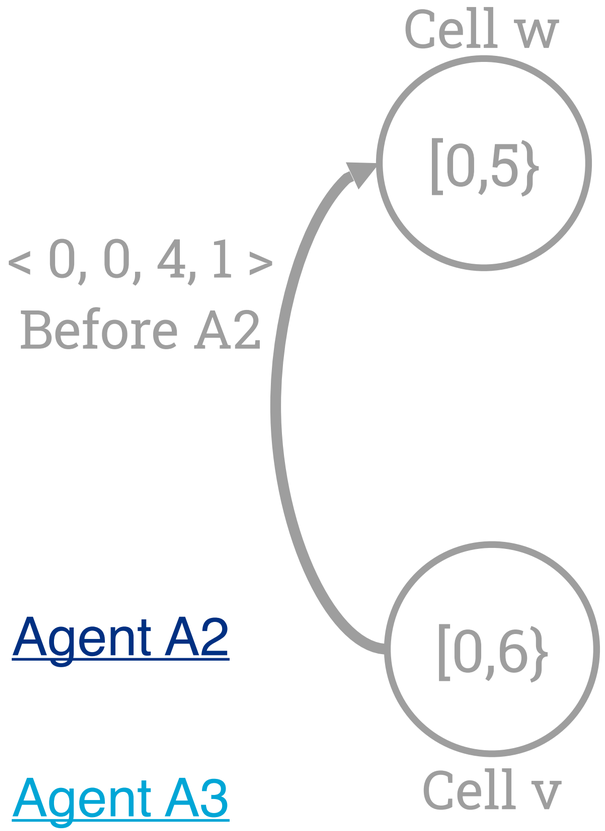




Multi-Agent Pathfinding

Constructing the SIPP graph





Edge Arrival Time Function

Start wait safe = 0

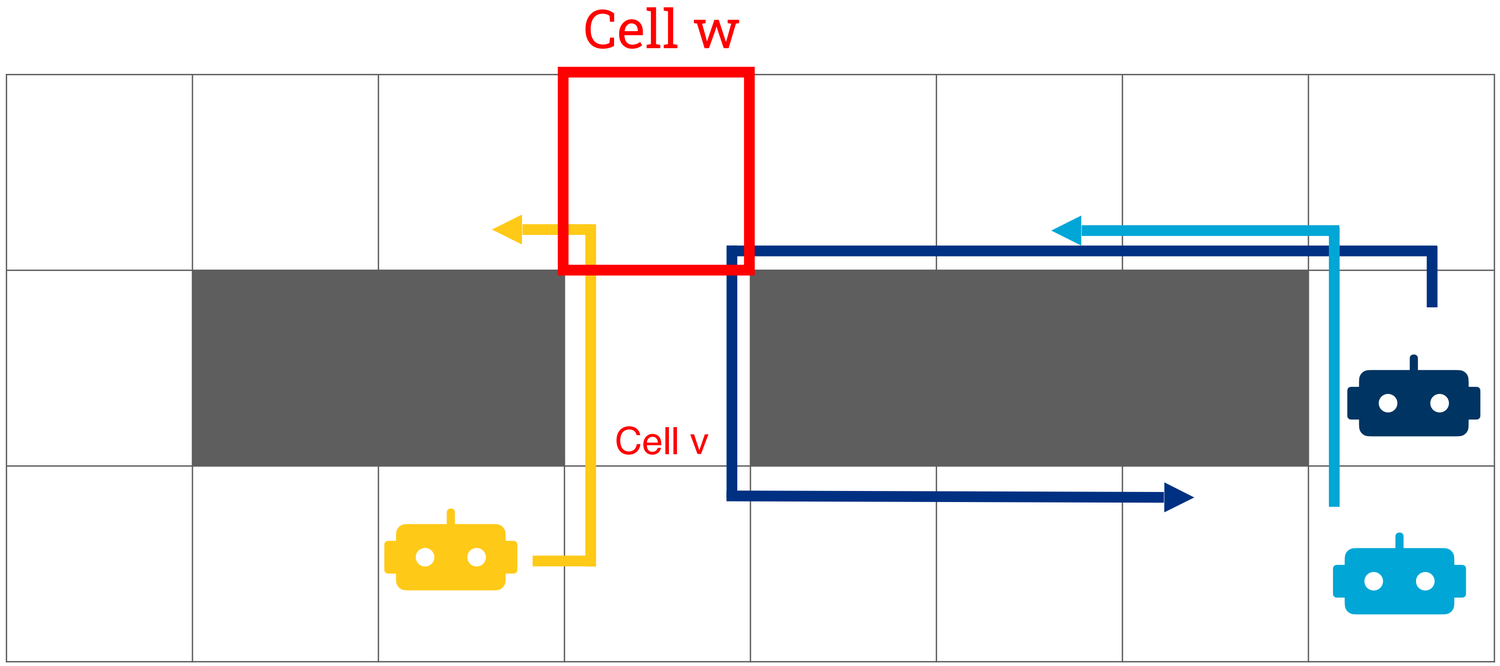
Start move = 0

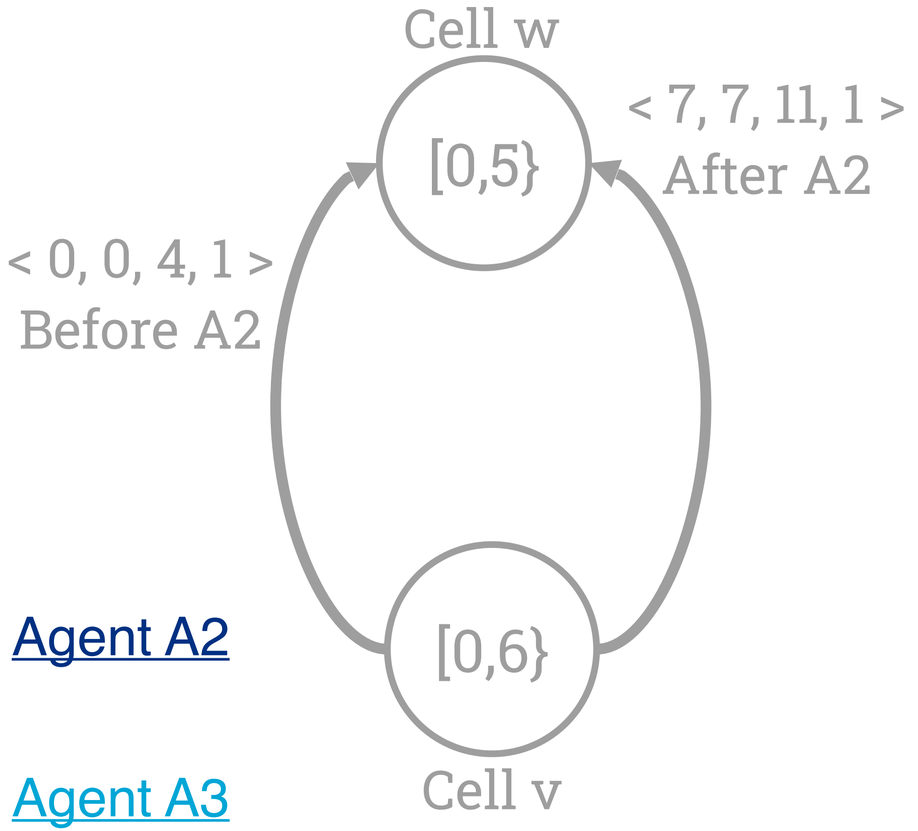
Move before = 4

Duration = 1

Multi-Agent Pathfinding

Constructing the SIPP graph





Edge Arrival Time Function

Start wait safe = 7

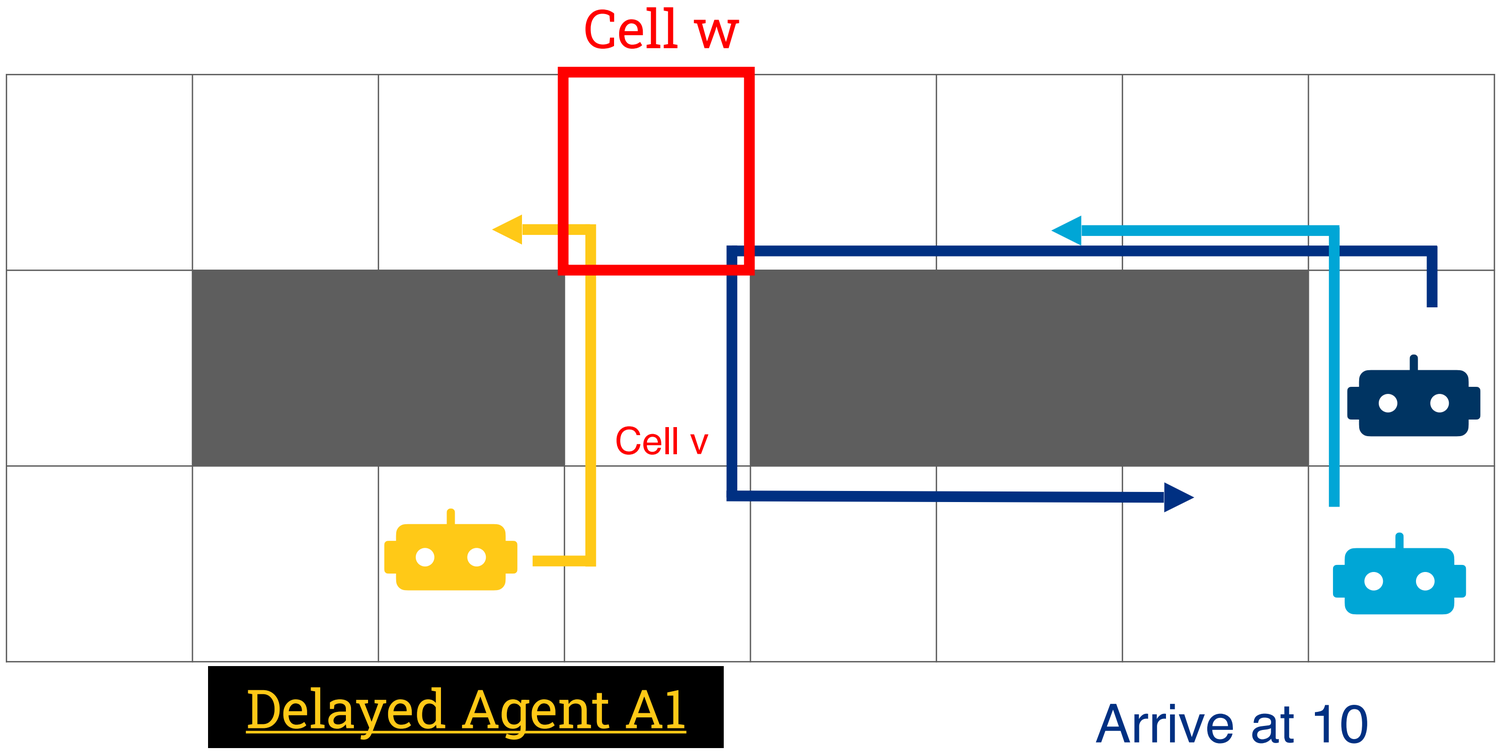
Start move = 7

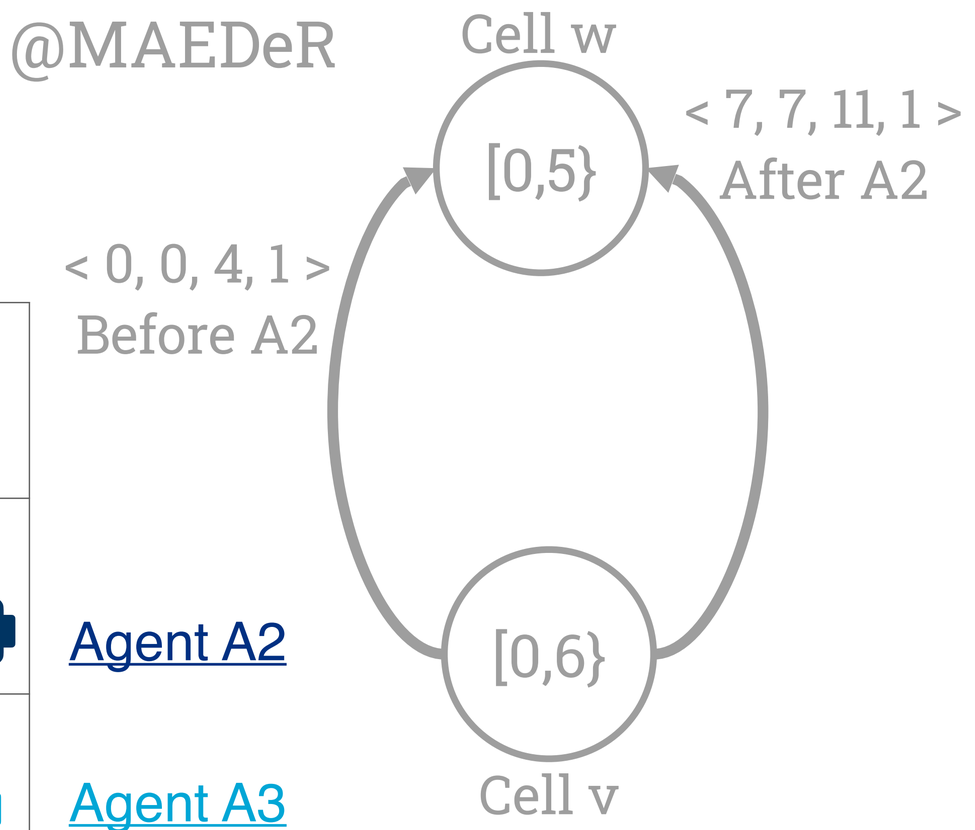
Move before = 11

Duration = 1

Multi-Agent Pathfinding

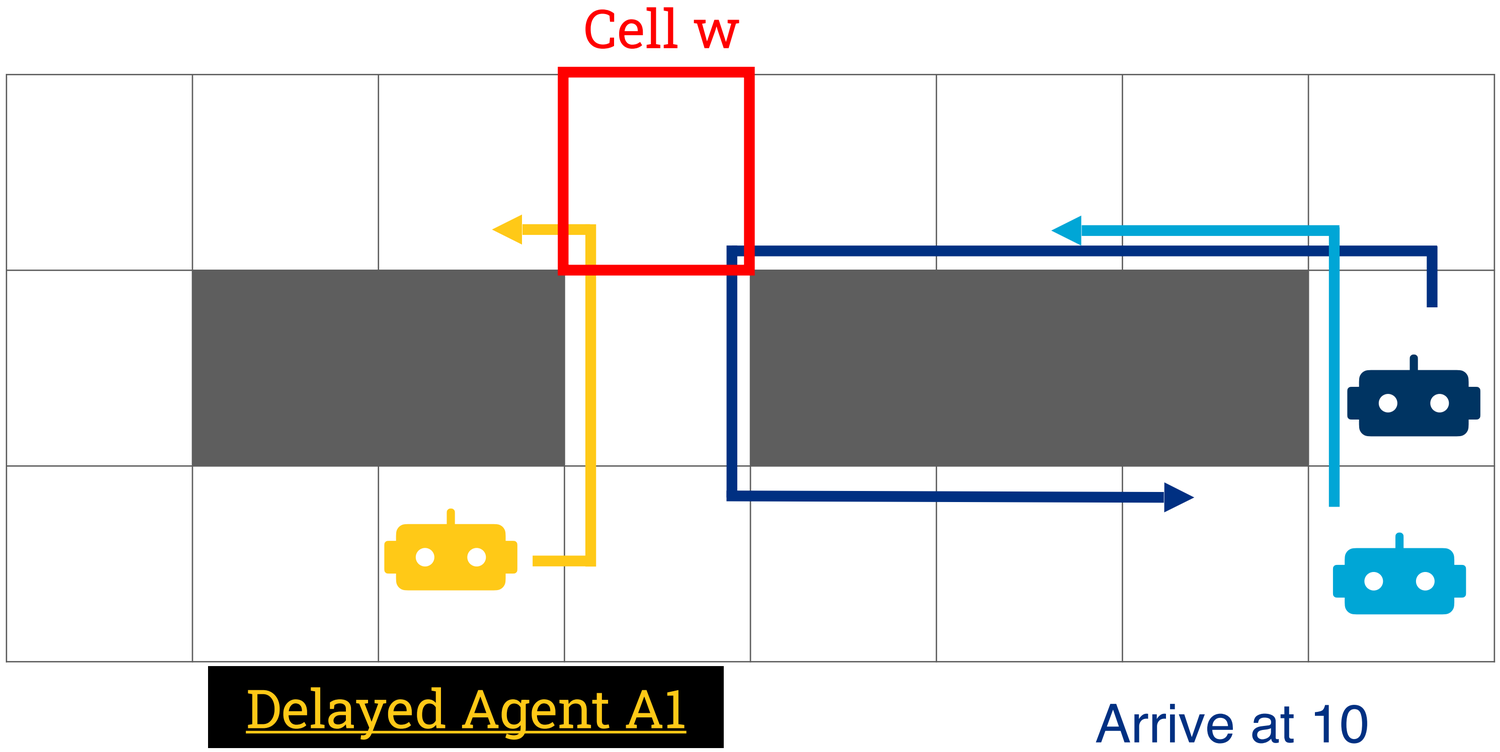
Constructing the SIPP graph





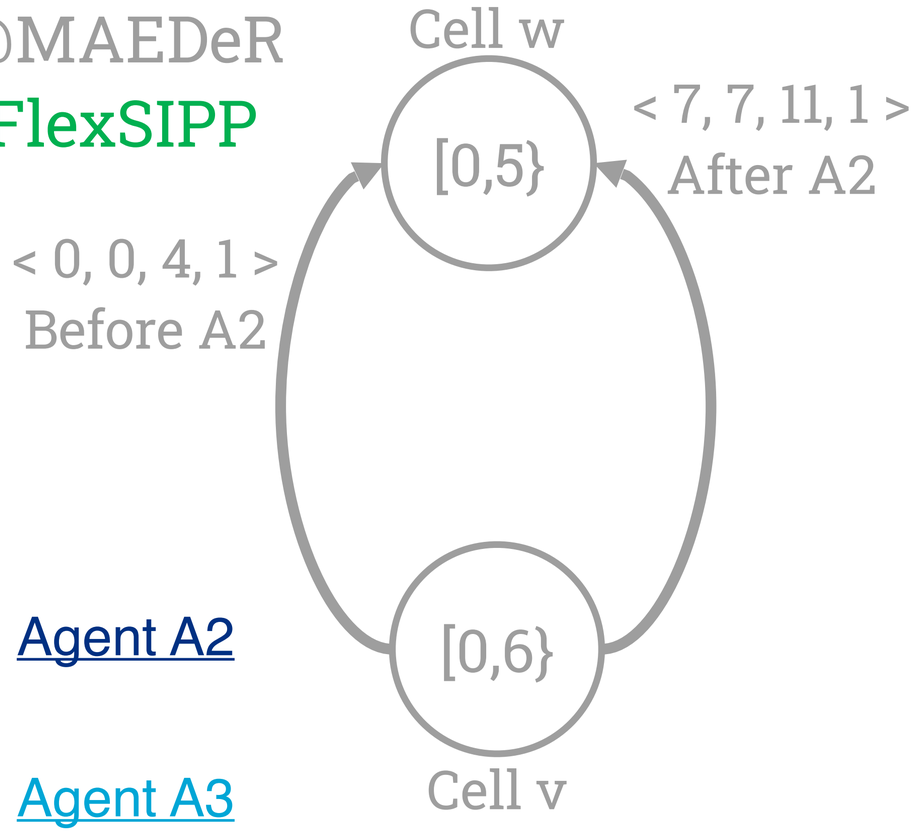
Multi-Agent Pathfinding

Constructing the SIPP graph



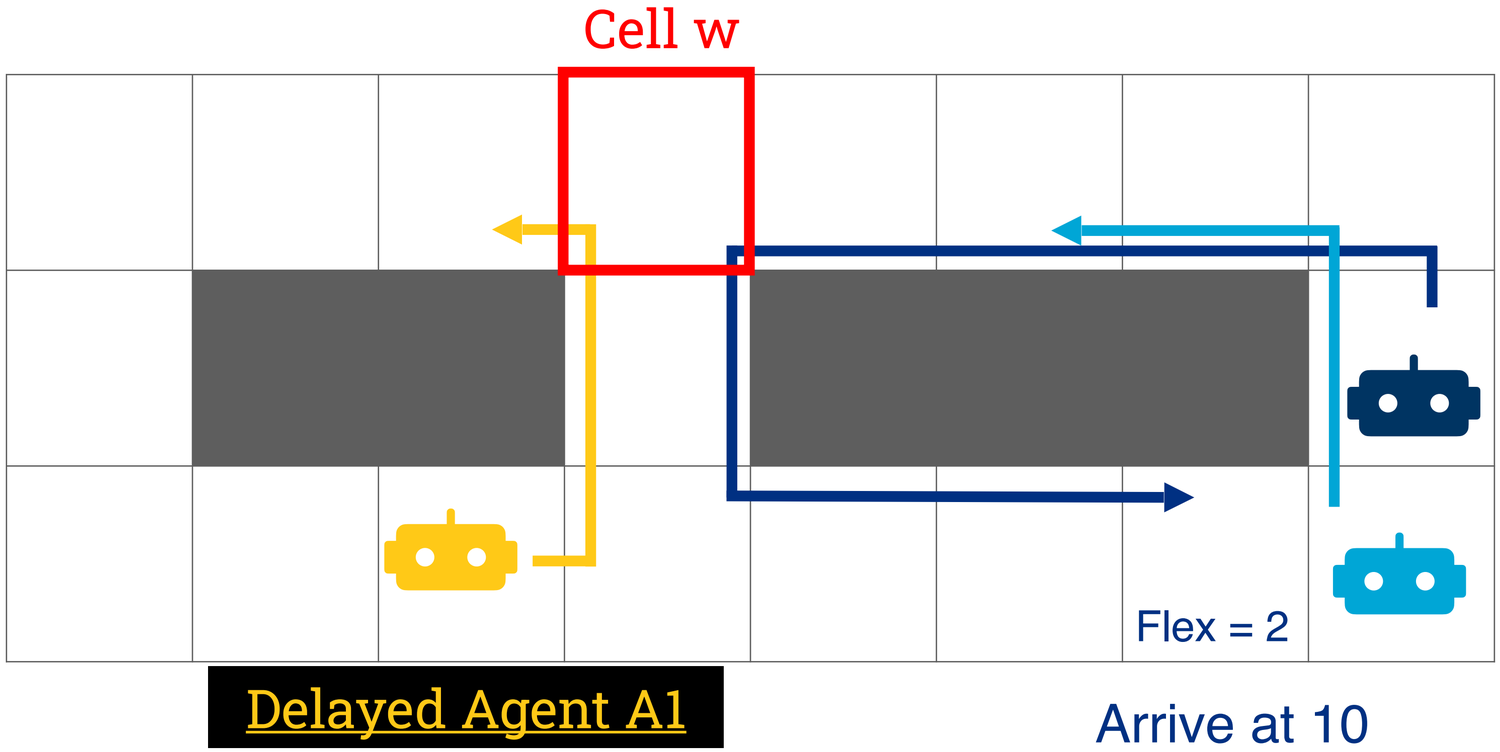
@MAEDeR

FlexSIPP



Multi-Agent Pathfinding

Constructing the SIPP graph



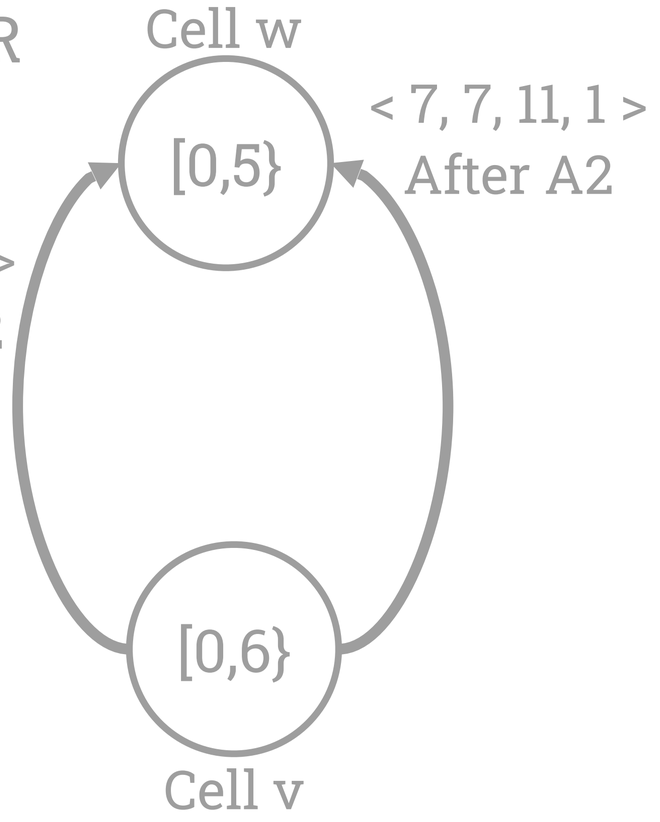
@MAEDeR

FlexSIPP

< 0, 0, 4, 1 >
Before A2

Agent A2

Agent A3

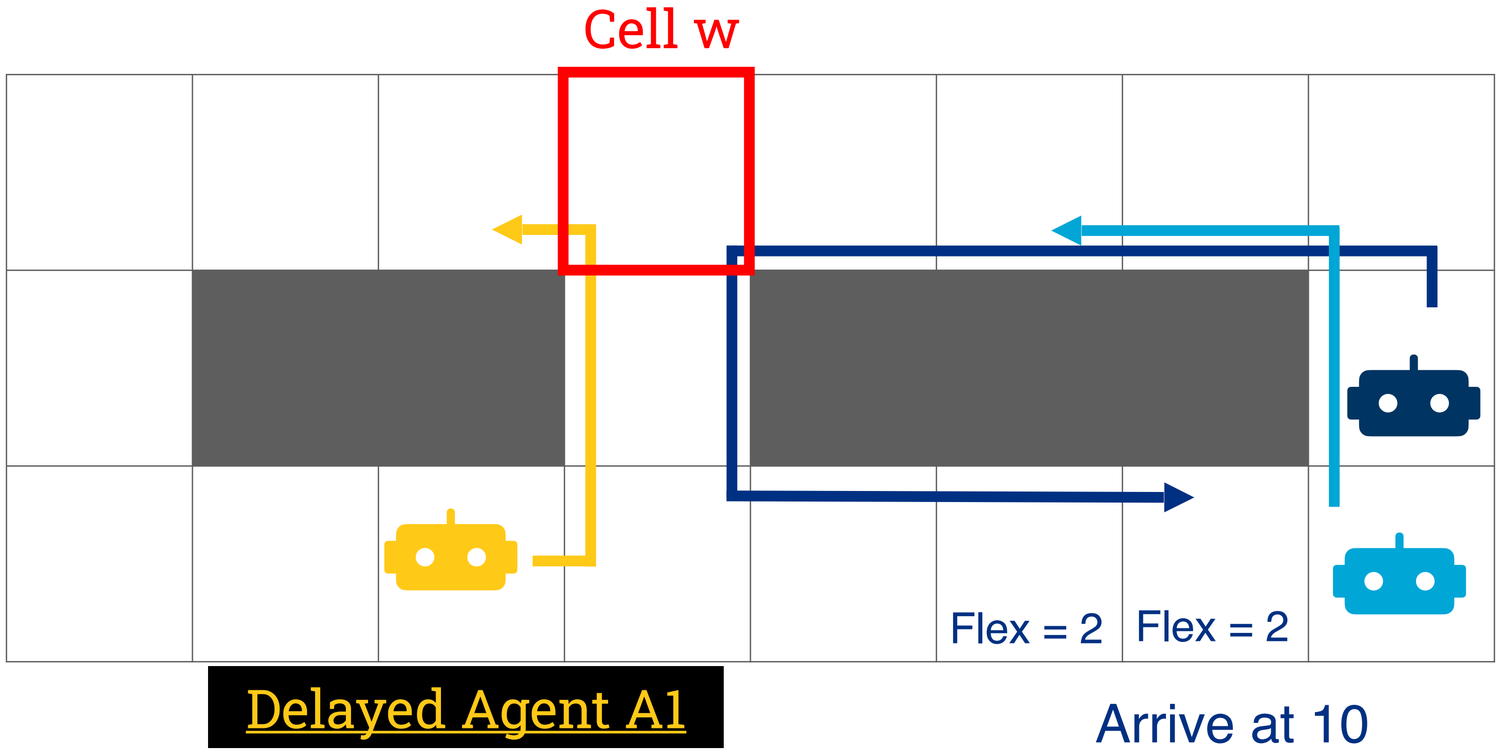


Multi-Agent Pathfinding

Constructing the SIPP graph

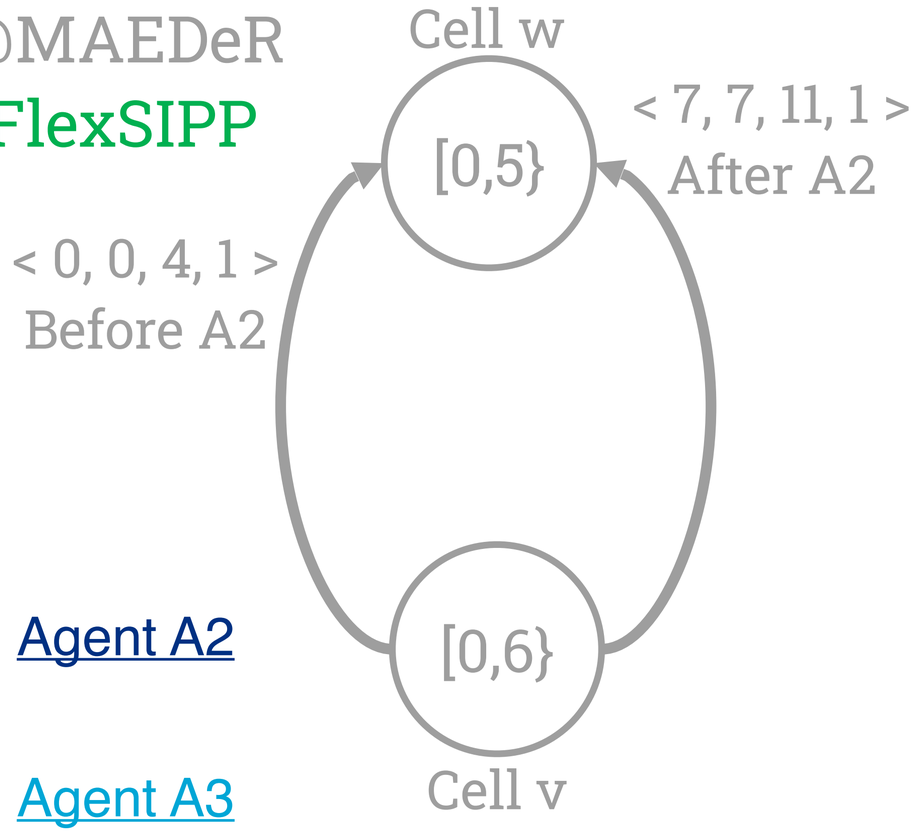
@MAEDeR

FlexSIPP



Agent A2

Agent A3

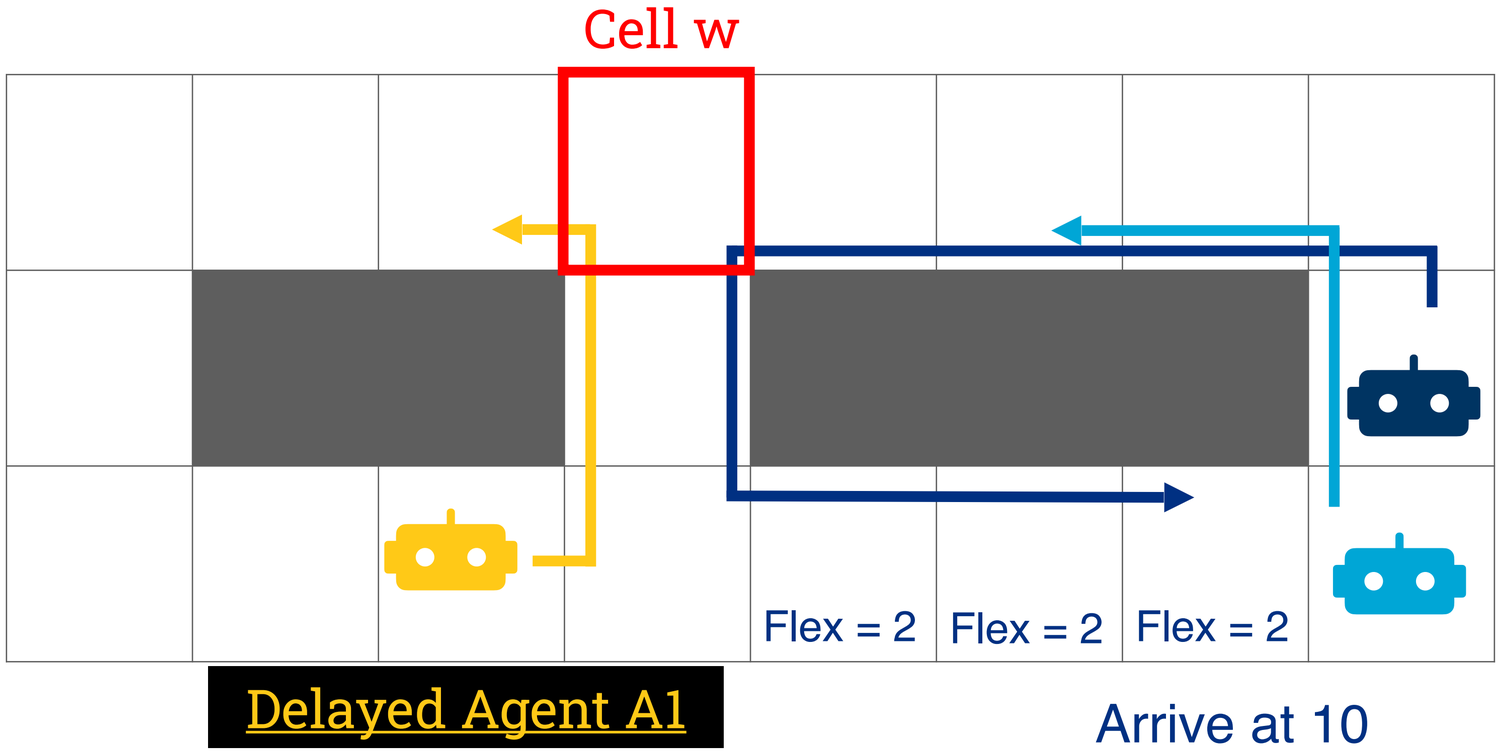


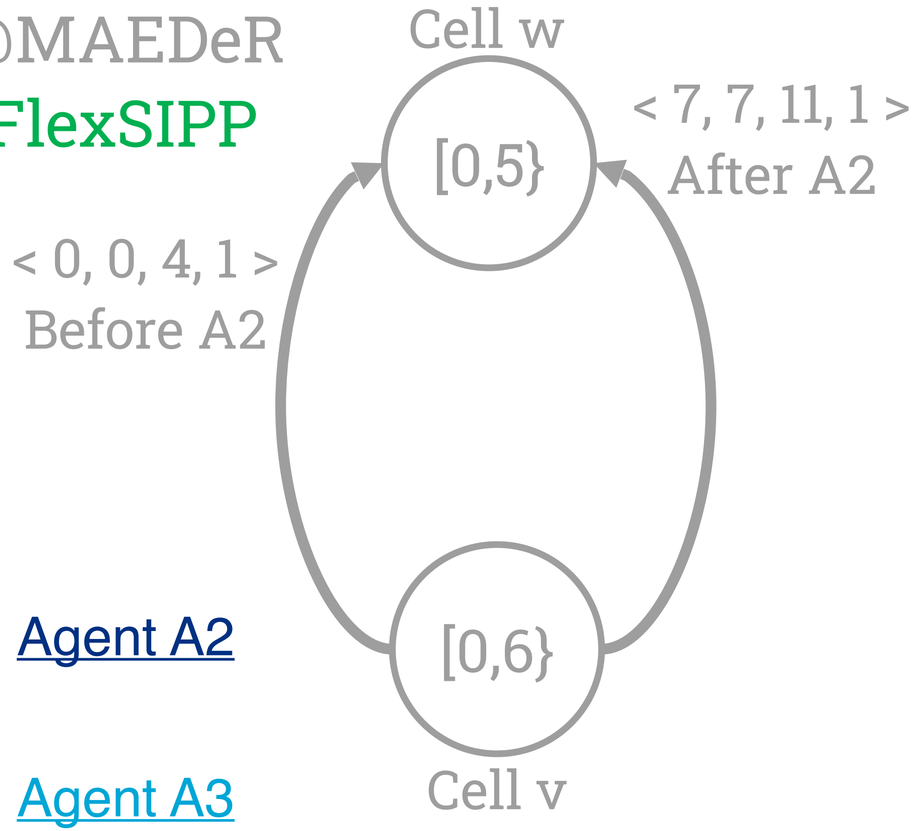
Multi-Agent Pathfinding

Constructing the SIPP graph

@MAEDeR

FlexSIPP



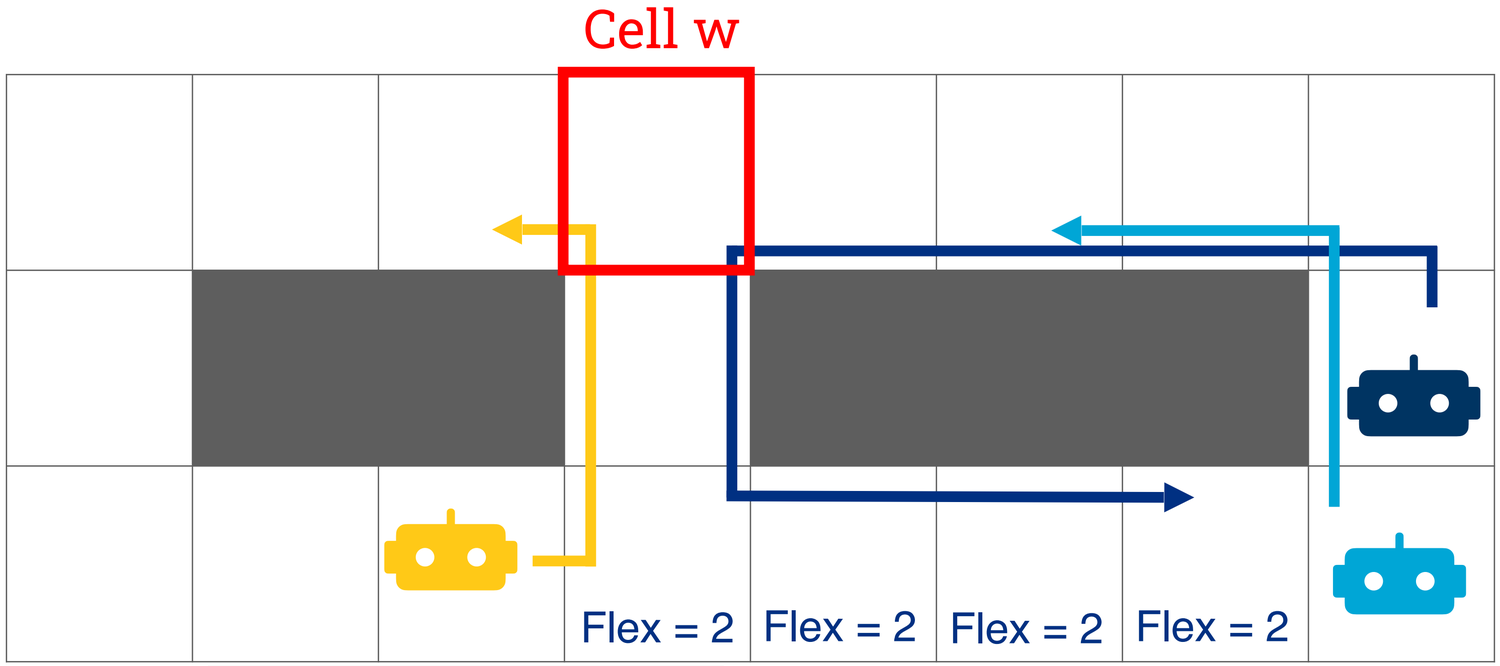


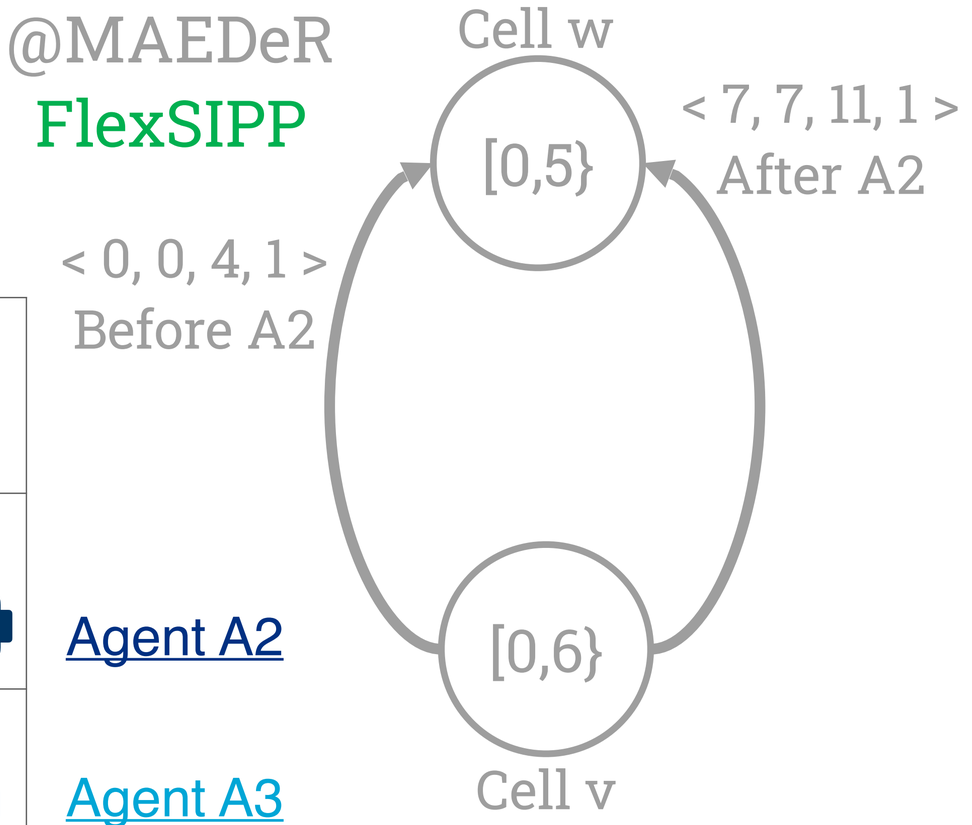
Multi-Agent Pathfinding

Constructing the SIPP graph

@MAEDeR

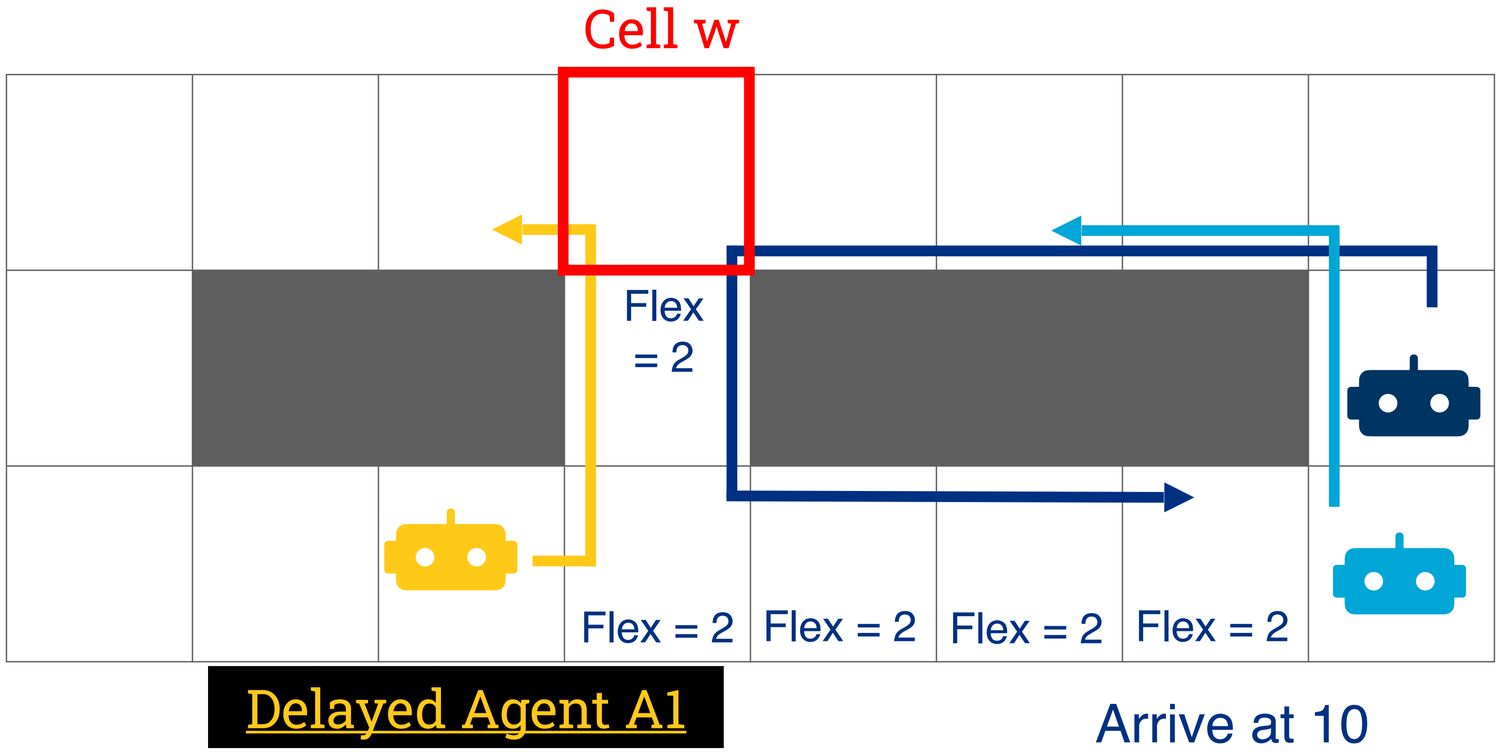
FlexSIPP





Multi-Agent Pathfinding

Constructing the SIPP graph



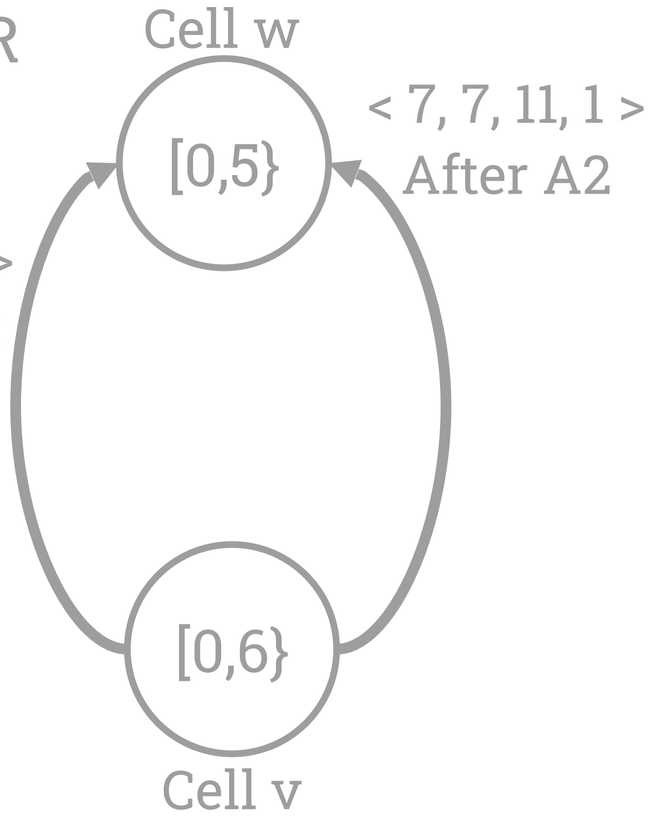
@MAEDeR

FlexSIPP

< 0, 0, 4, 1 >
Before A2

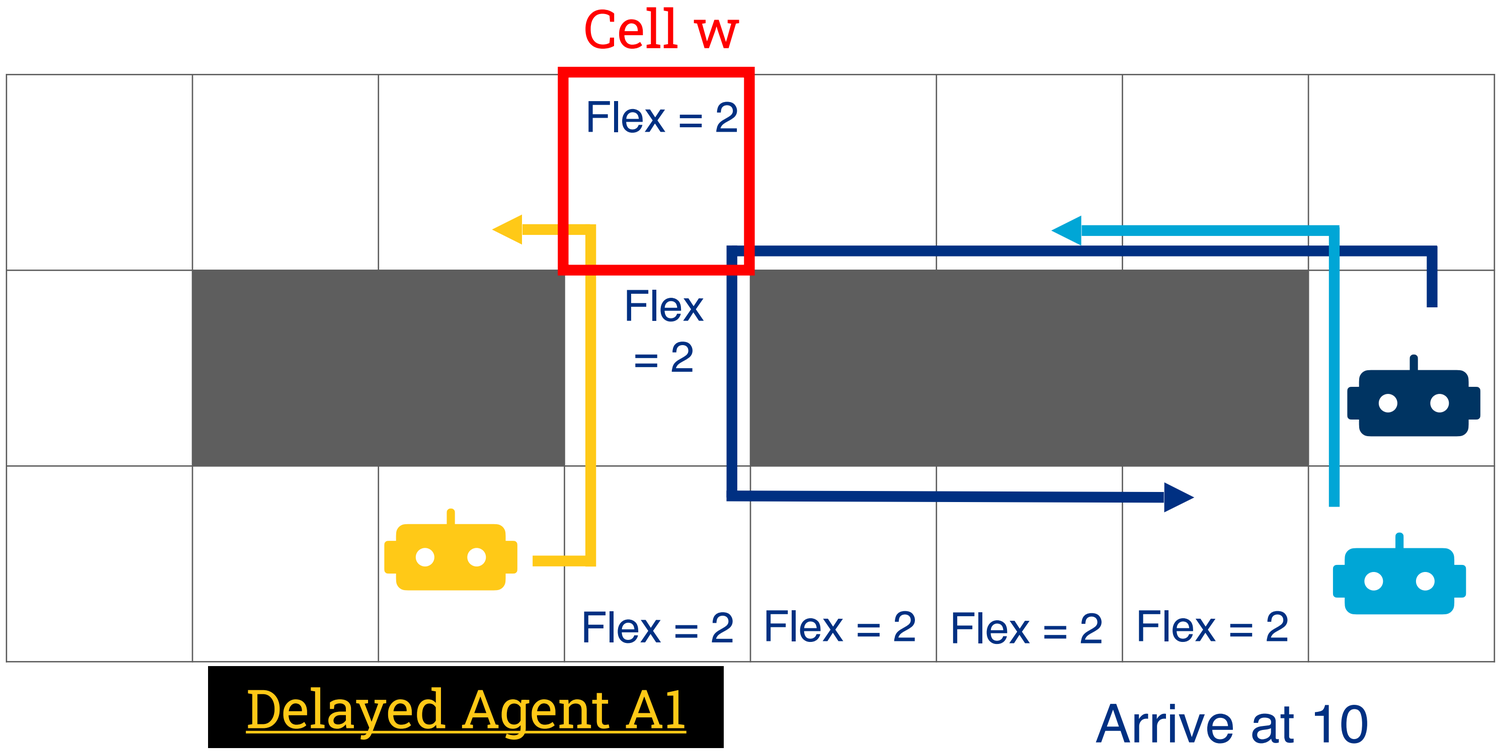
Agent A2

Agent A3



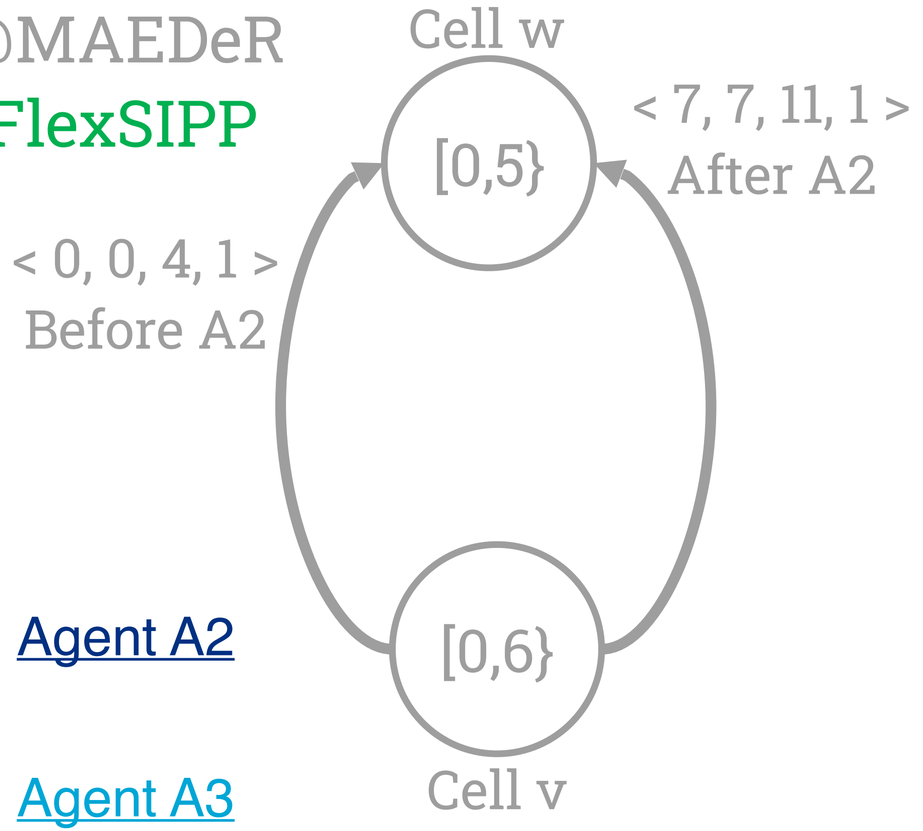
Multi-Agent Pathfinding

Constructing the SIPP graph



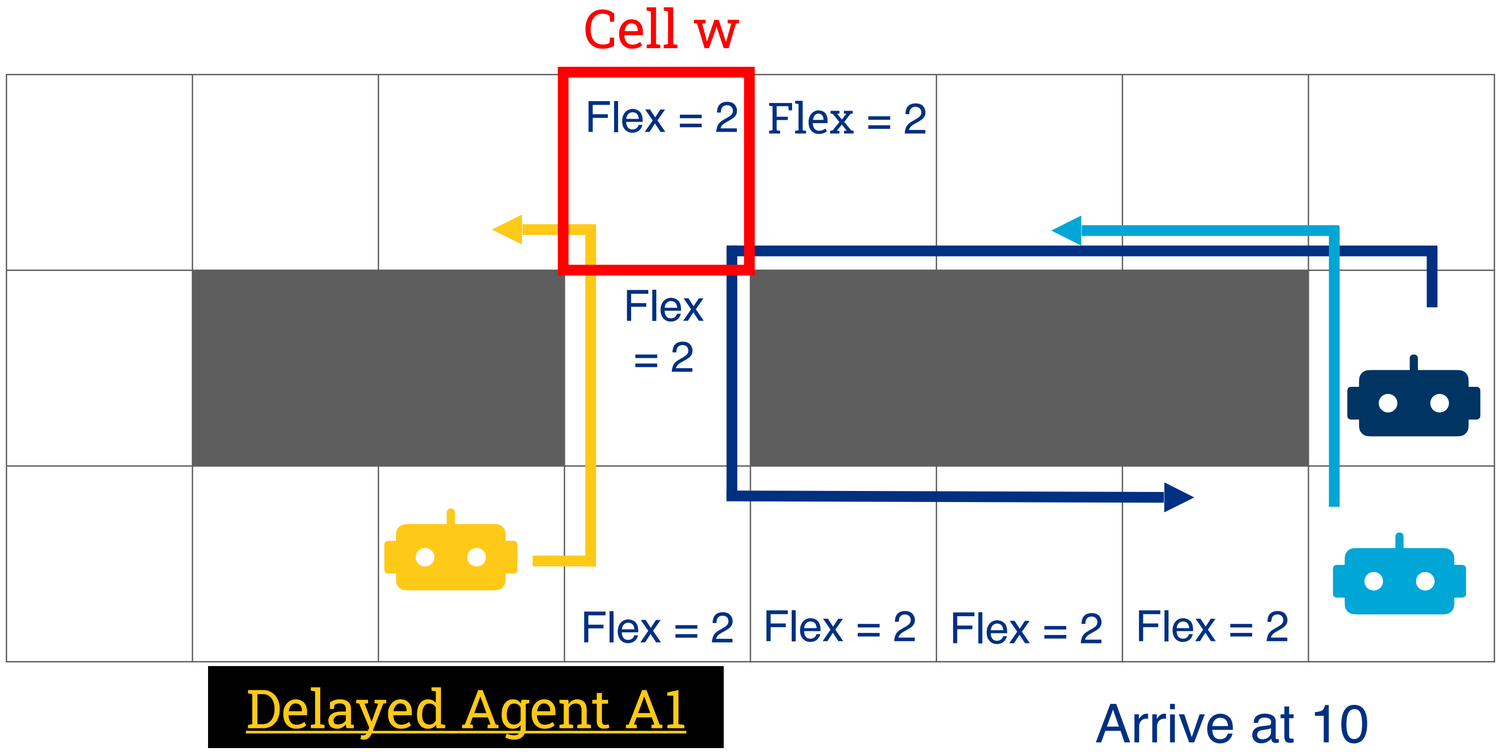
@MAEDeR

FlexSIPP



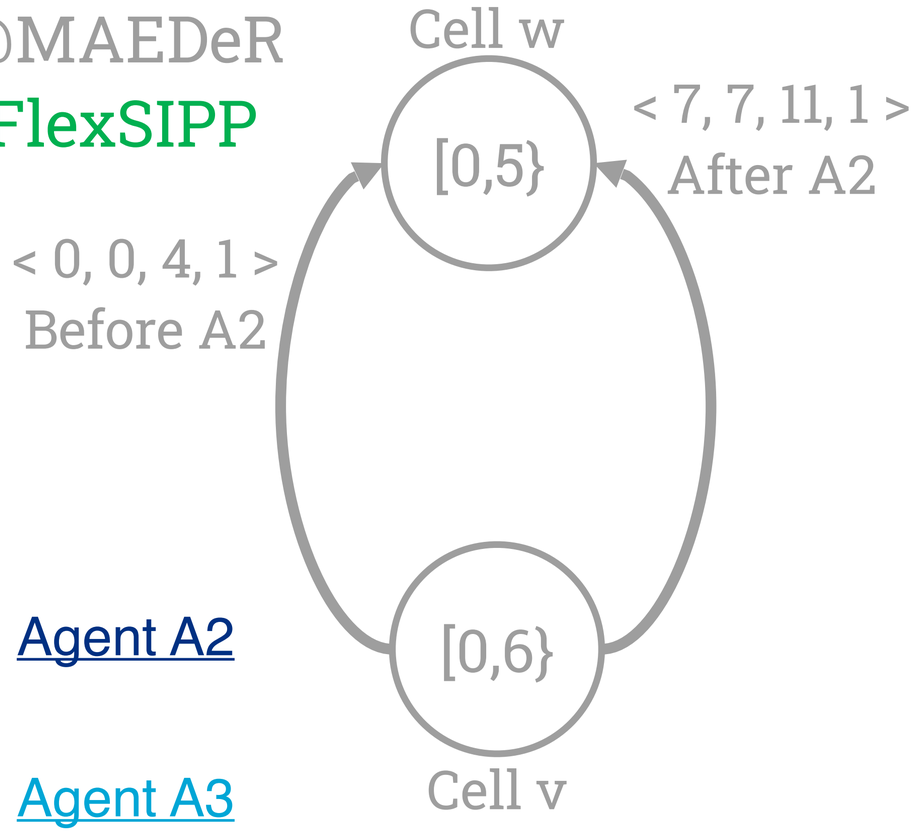
Multi-Agent Pathfinding

Constructing the SIPP graph



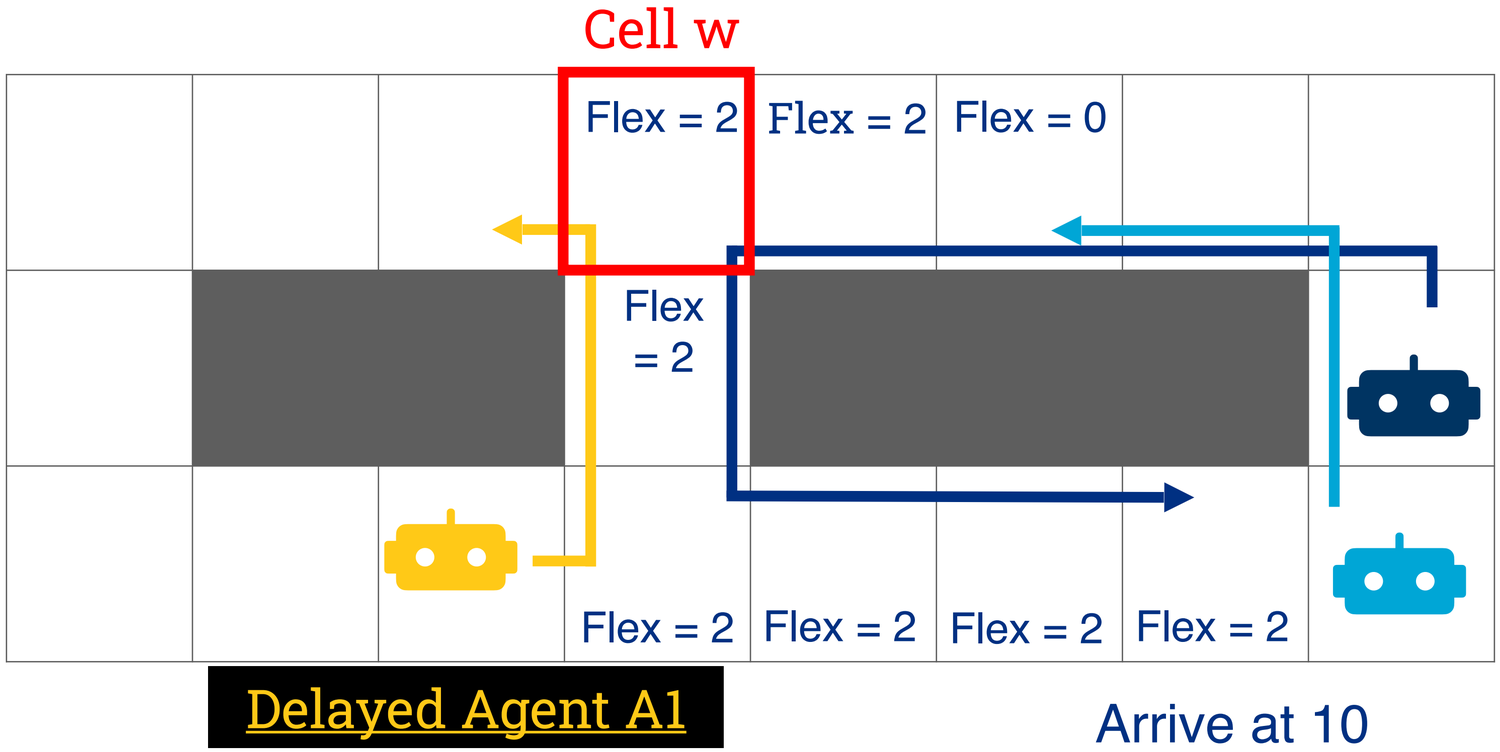
@MAEDeR

FlexSIPP



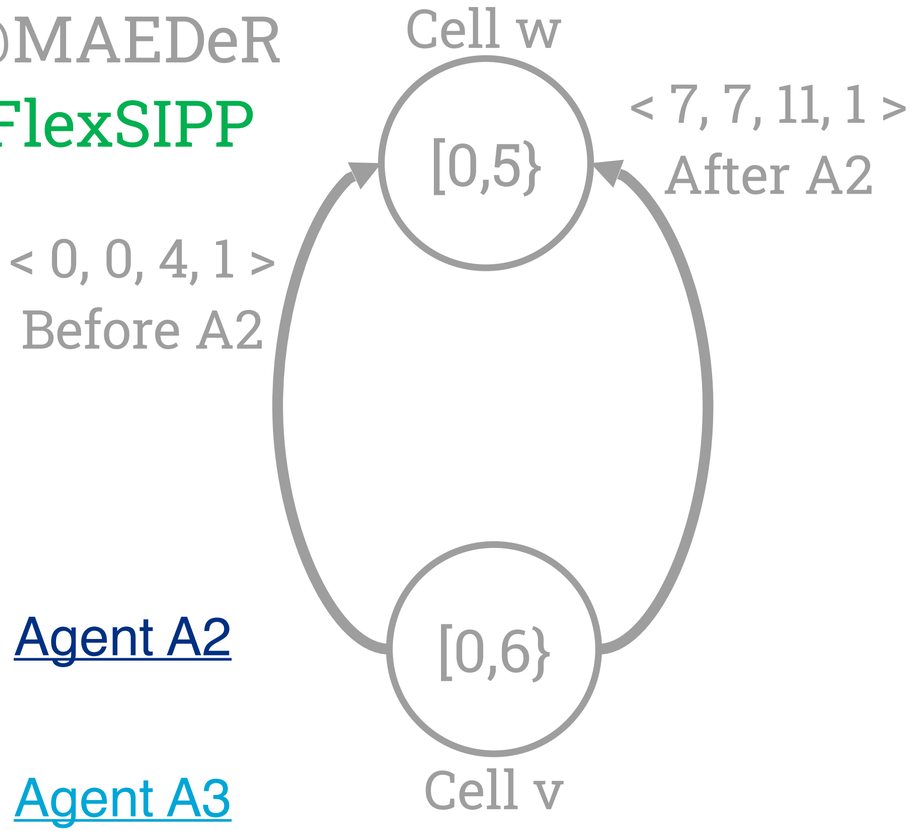
Multi-Agent Pathfinding

Constructing the SIPP graph



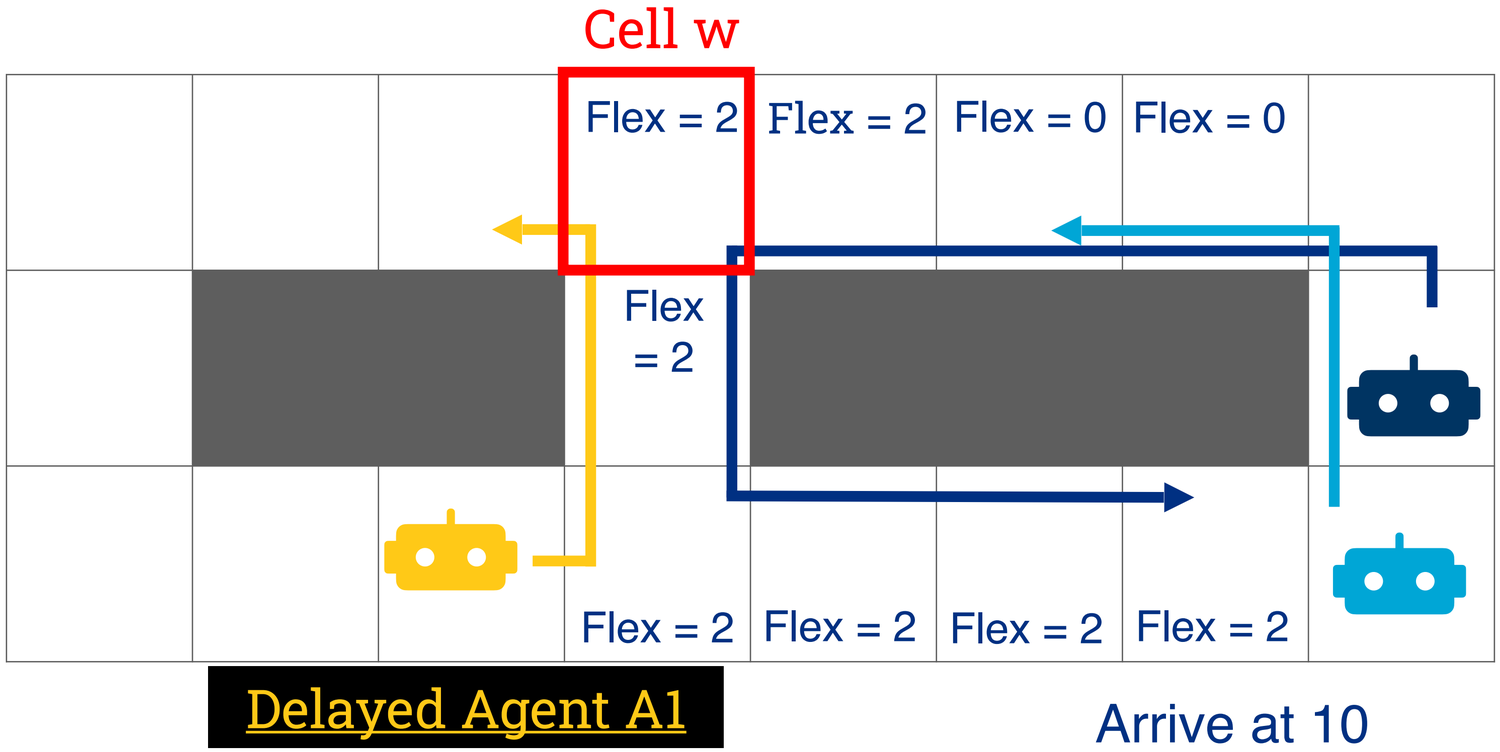
@MAEDeR

FlexSIPP



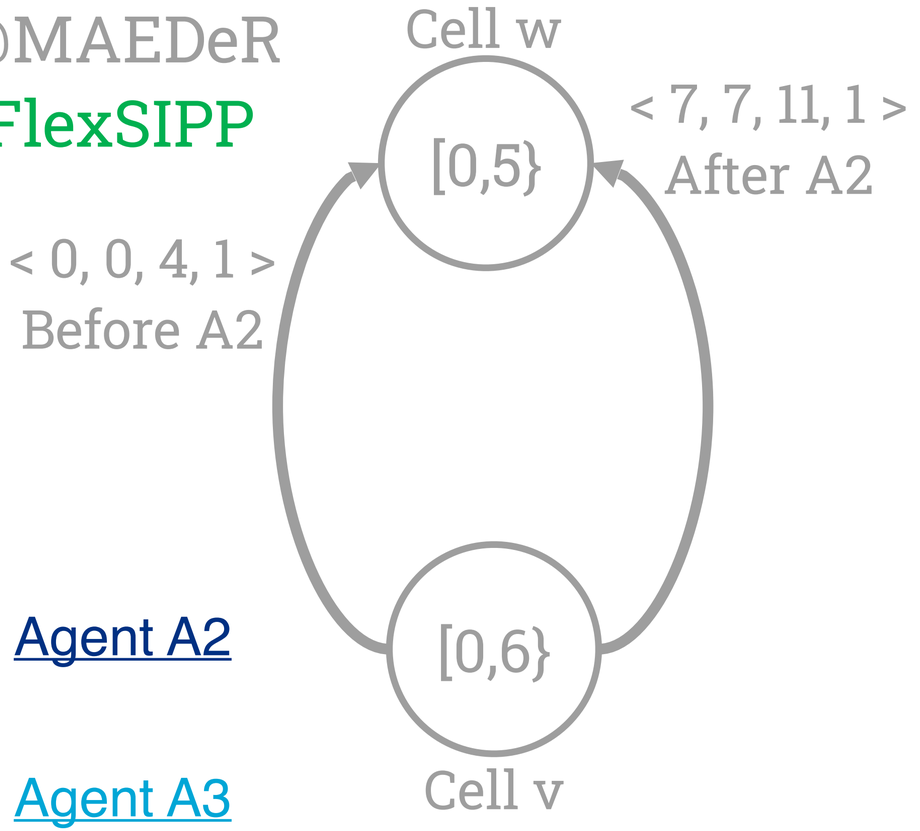
Multi-Agent Pathfinding

Constructing the SIPP graph



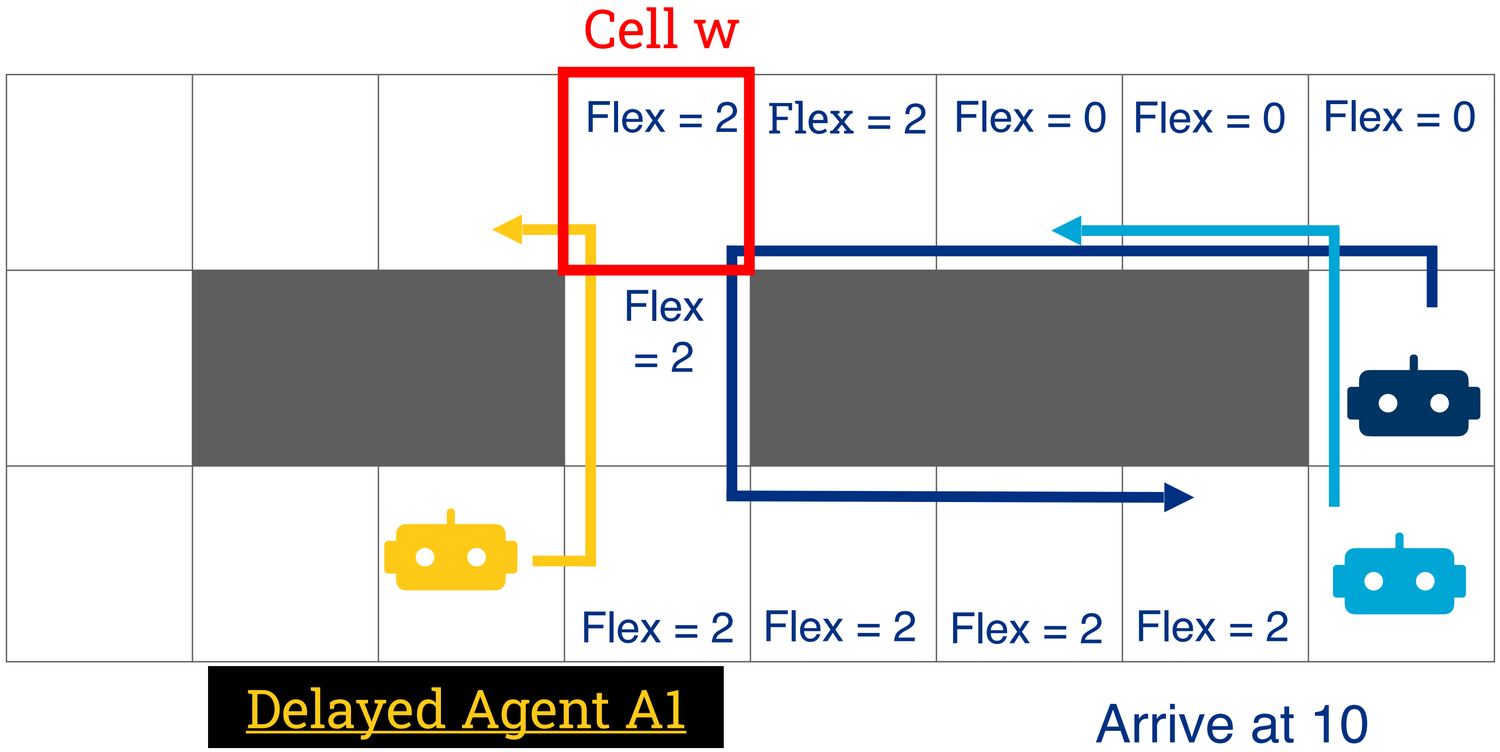
@MAEDeR

FlexSIPP



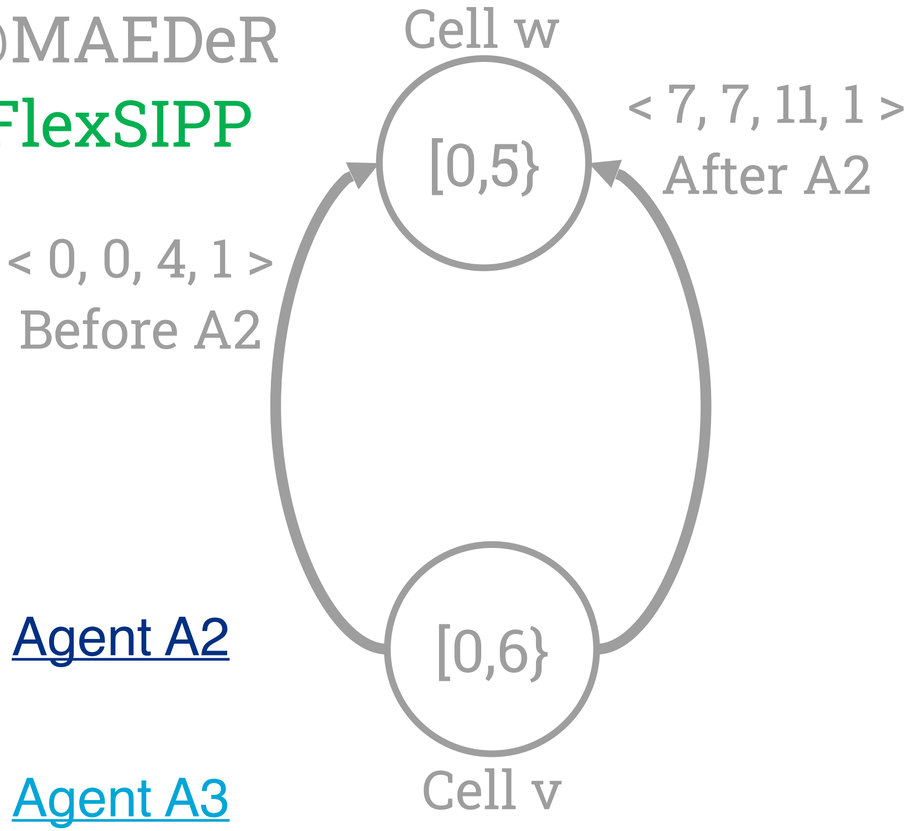
Multi-Agent Pathfinding

Constructing the SIPP graph



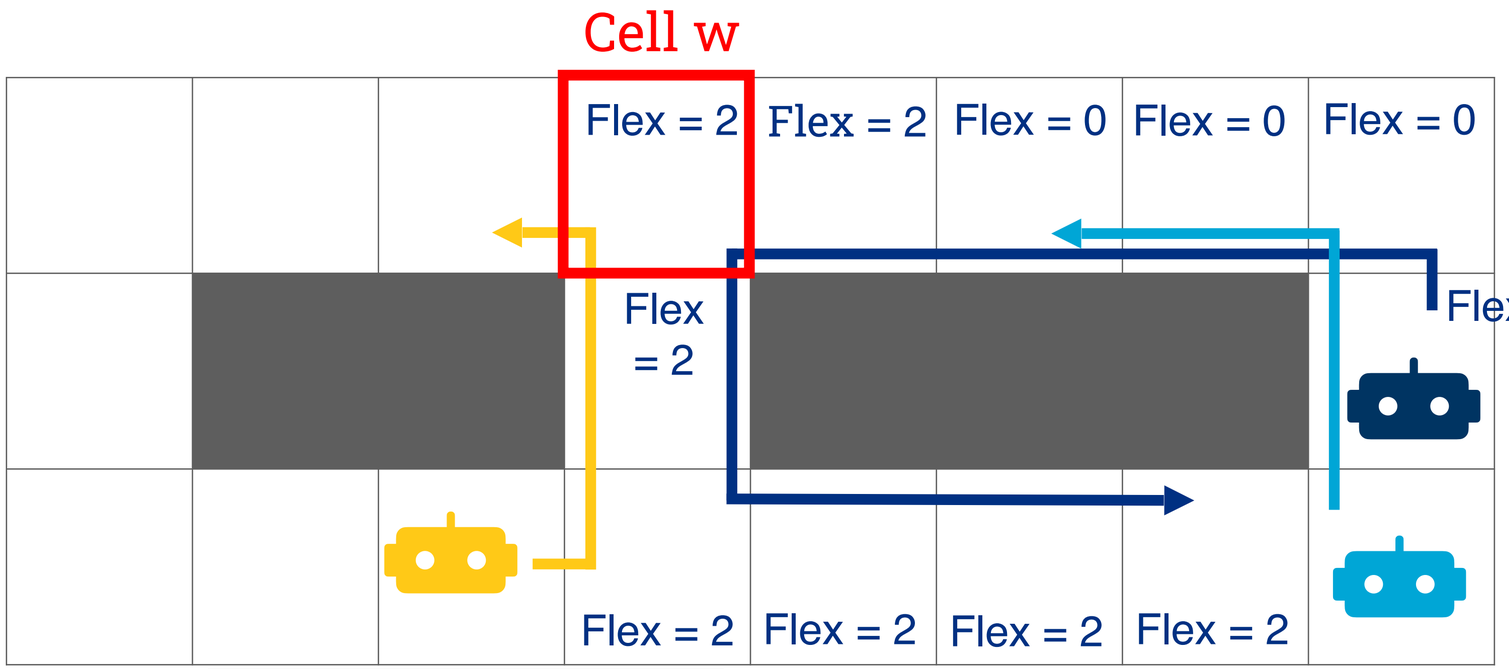
@MAEDeR

FlexSIPP



Multi-Agent Pathfinding

Constructing the SIPP graph



Delayed Agent A1

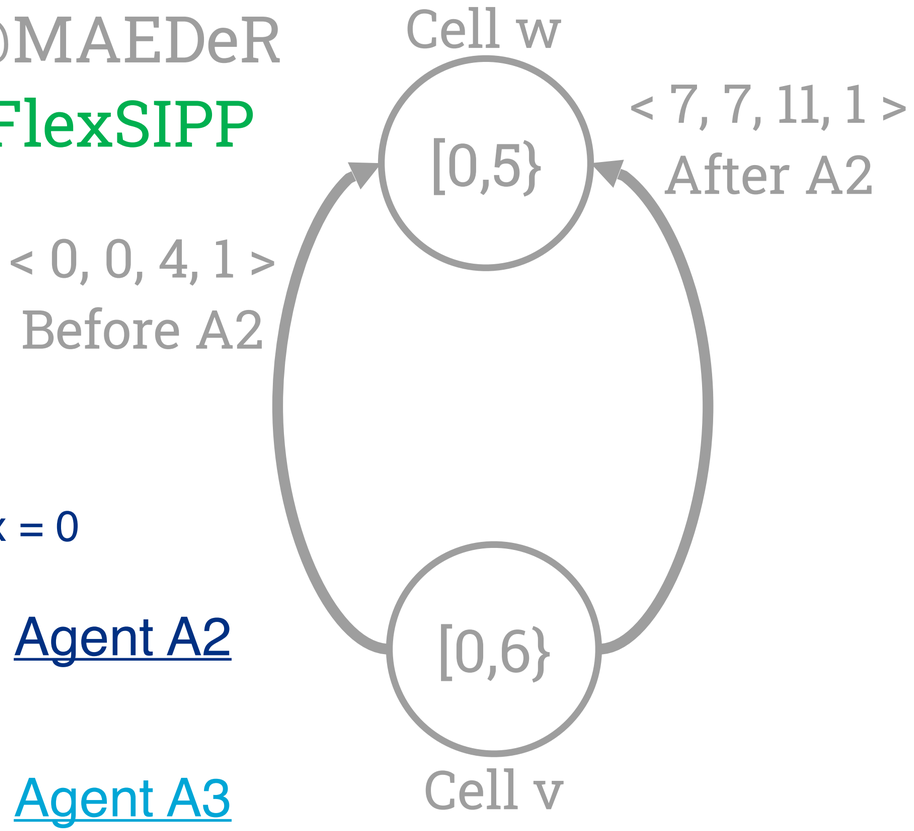
Agent A2

Agent A3

Arrive at 10
Deadline at 12

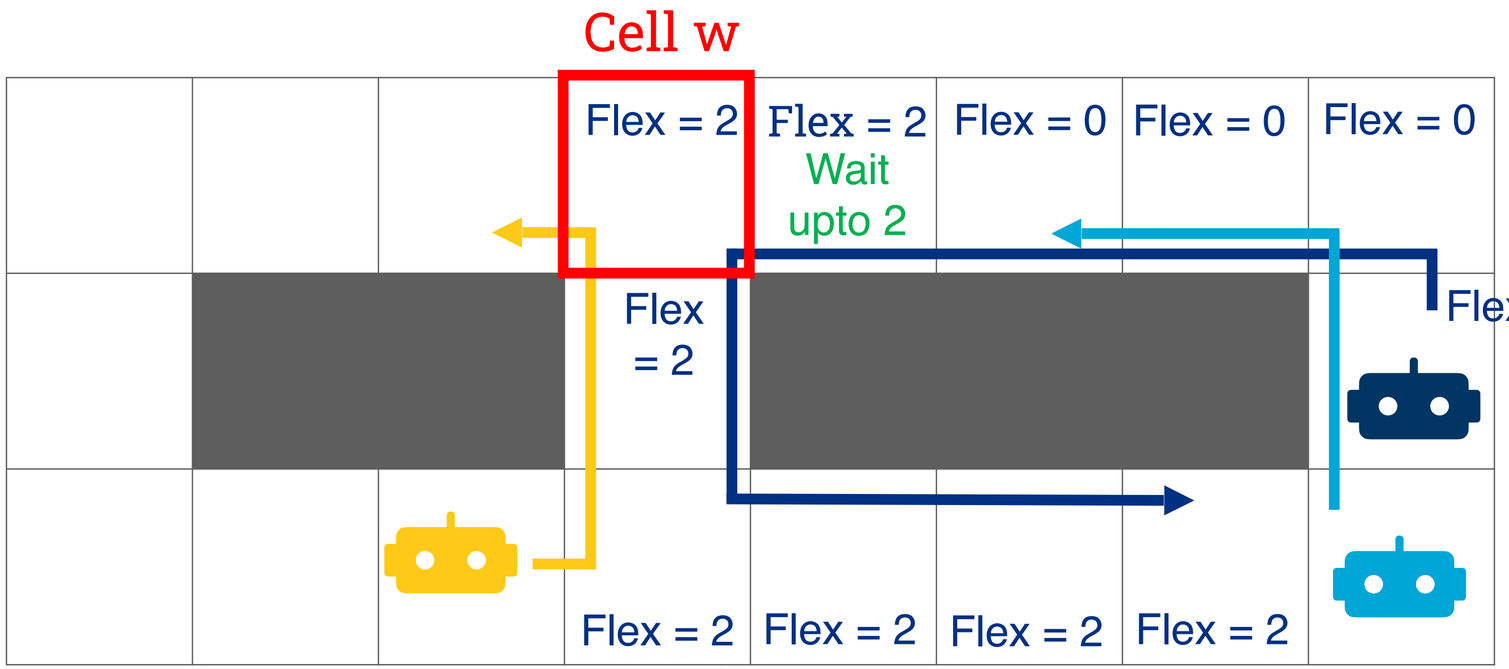
@MAEDeR

FlexSIPP



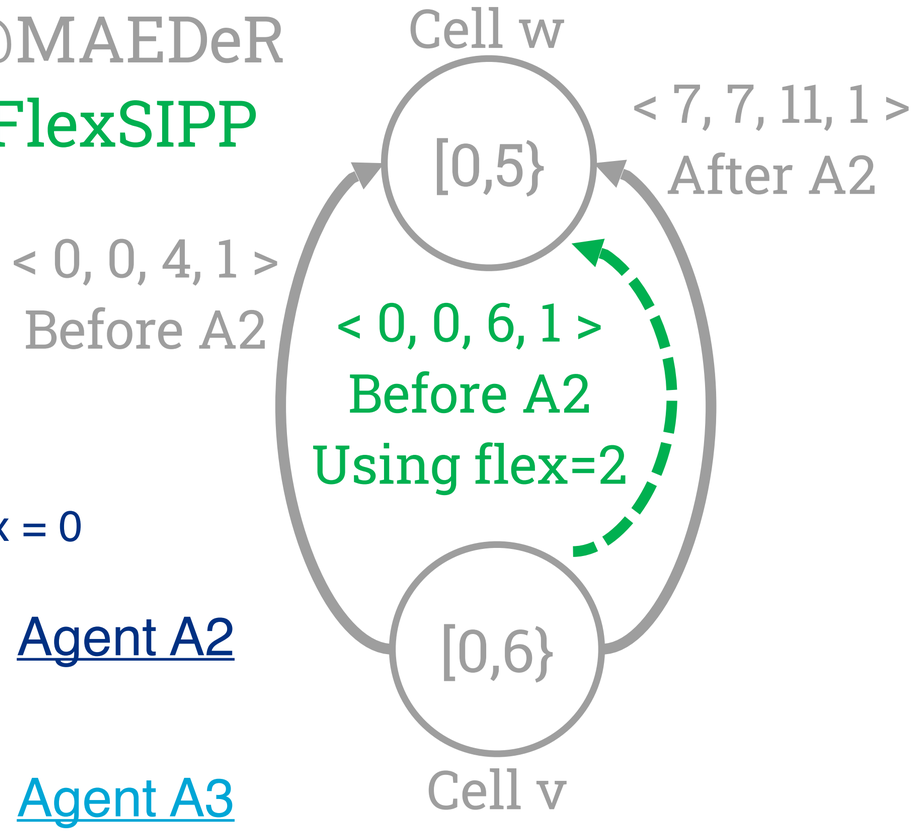
Multi-Agent Pathfinding

Constructing the SIPP graph

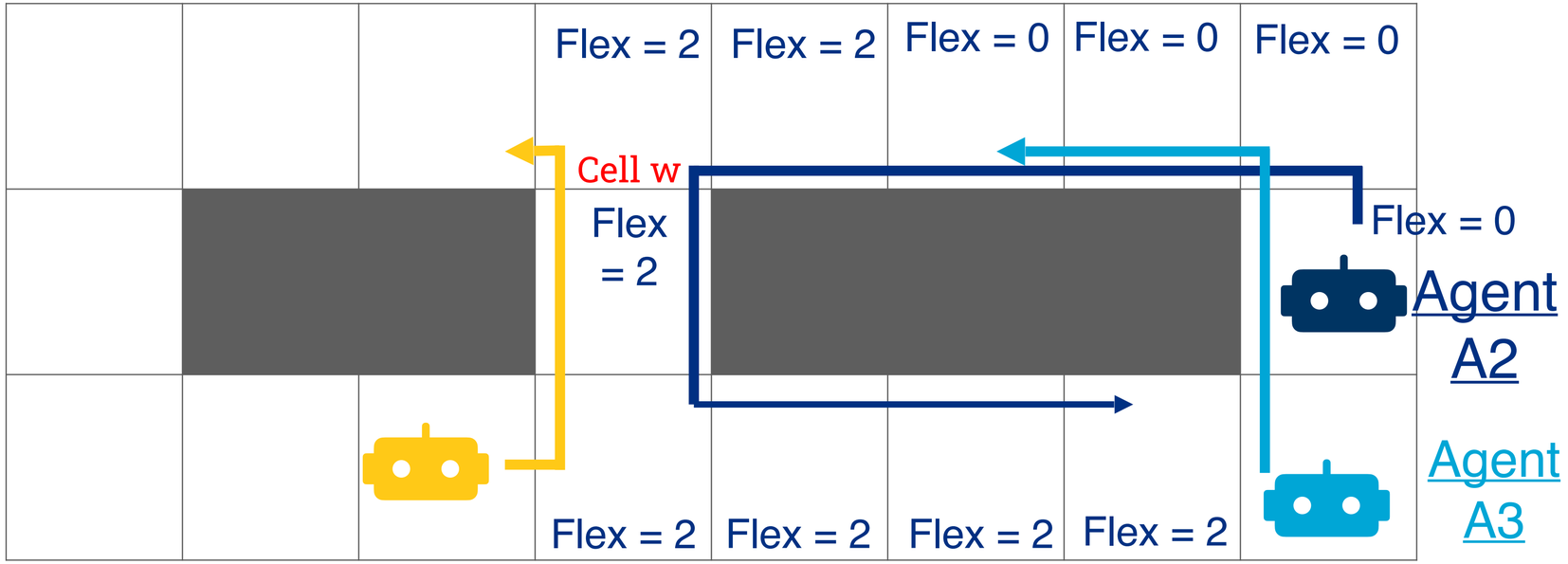


Arrive at 10
Deadline at 12

@MAEDeR
FlexSIPP



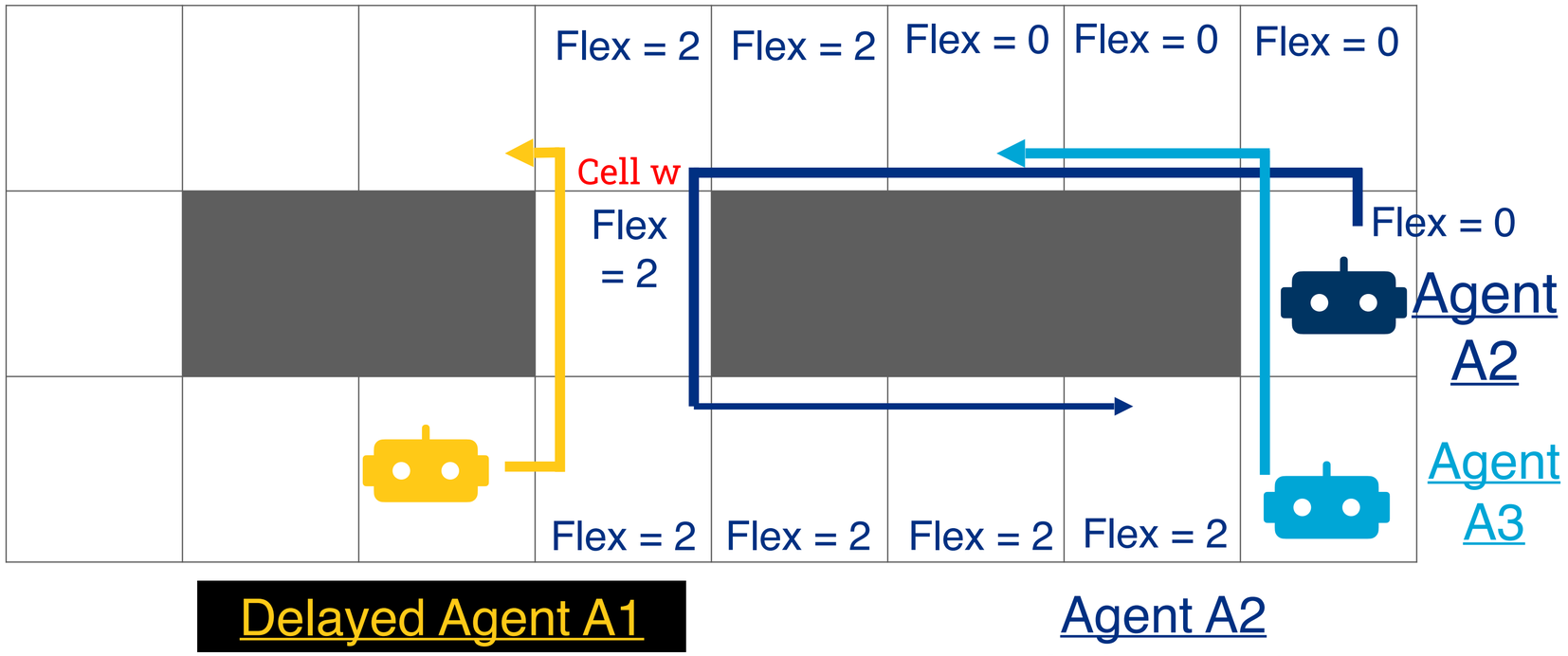
Any-Start-Time Plan

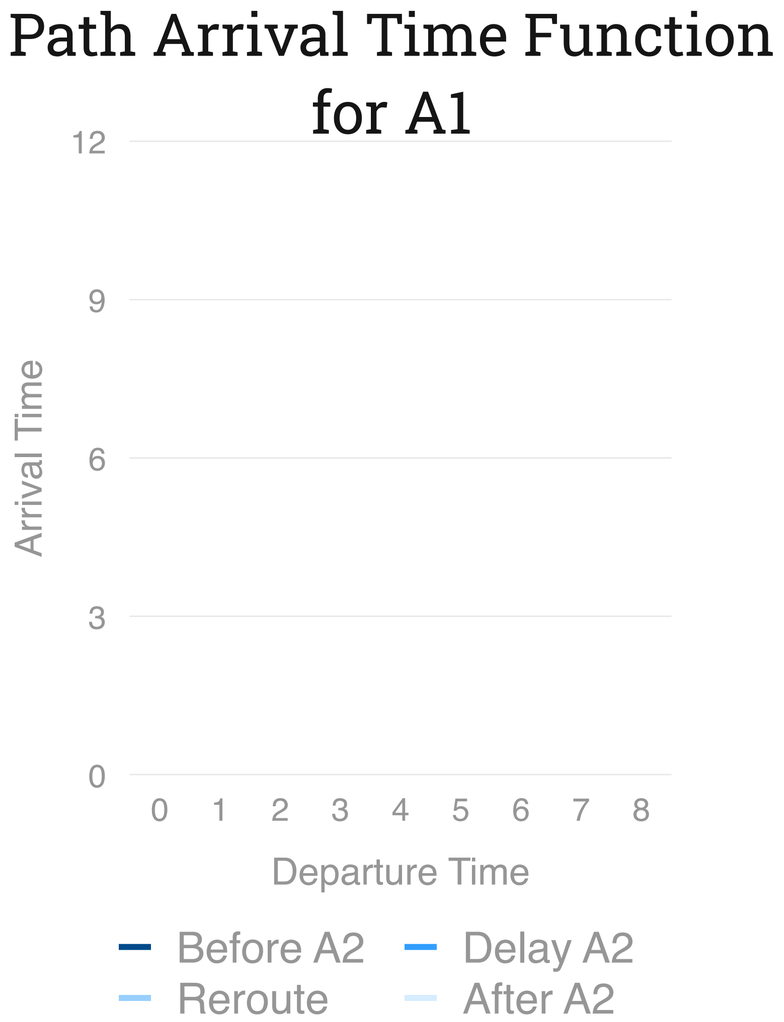


Delayed Agent A1

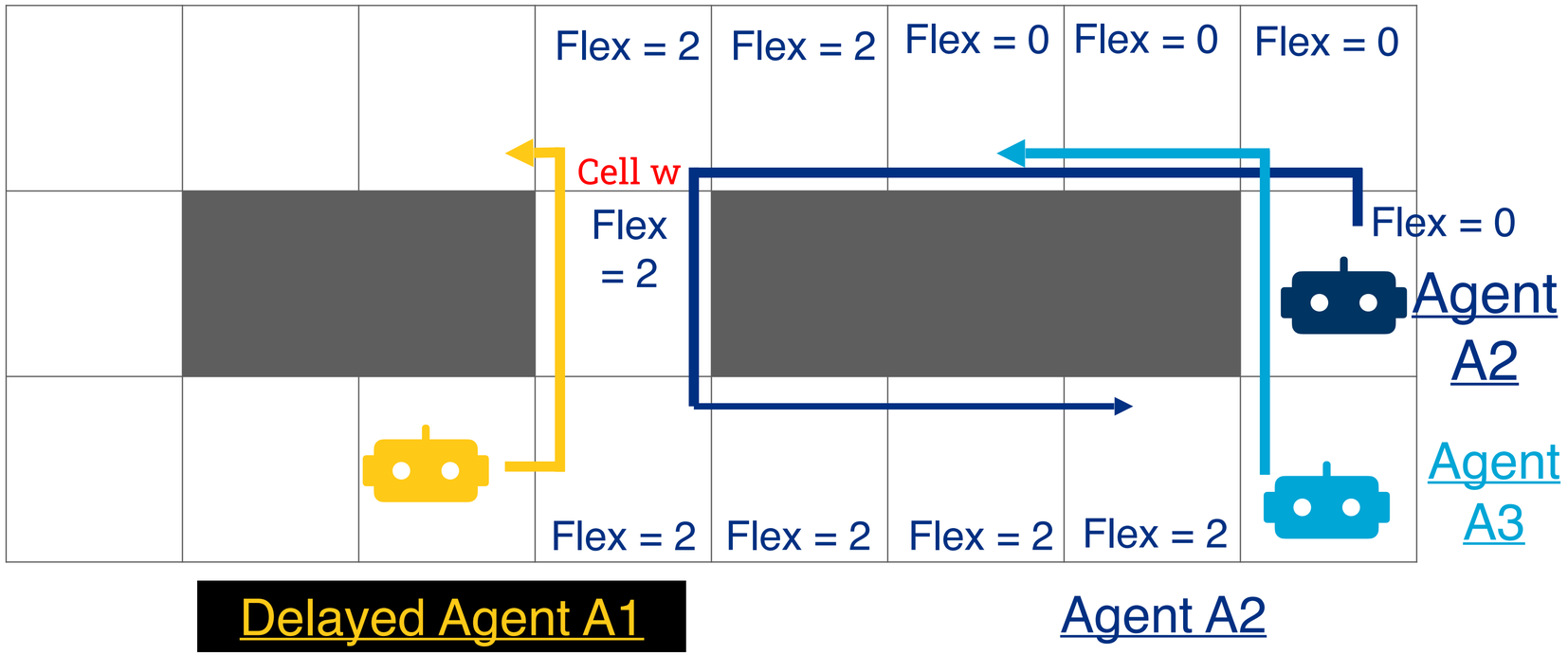
Agent A2
Arrive at 10
Deadline at 12

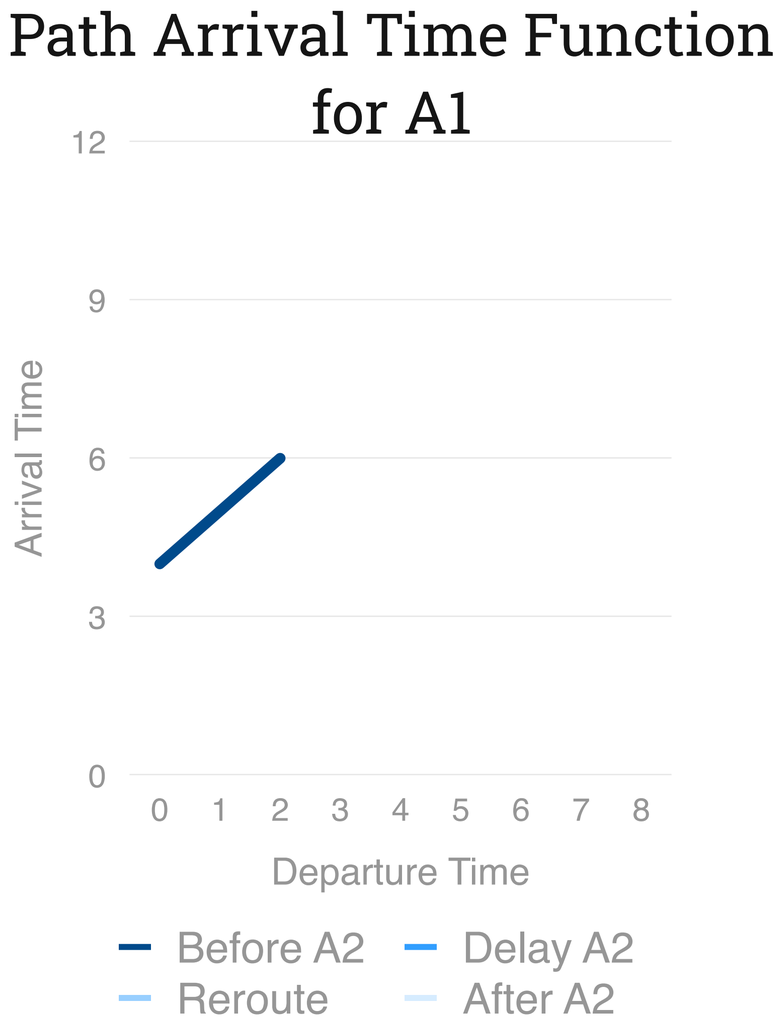
Any-Start-Time Plan



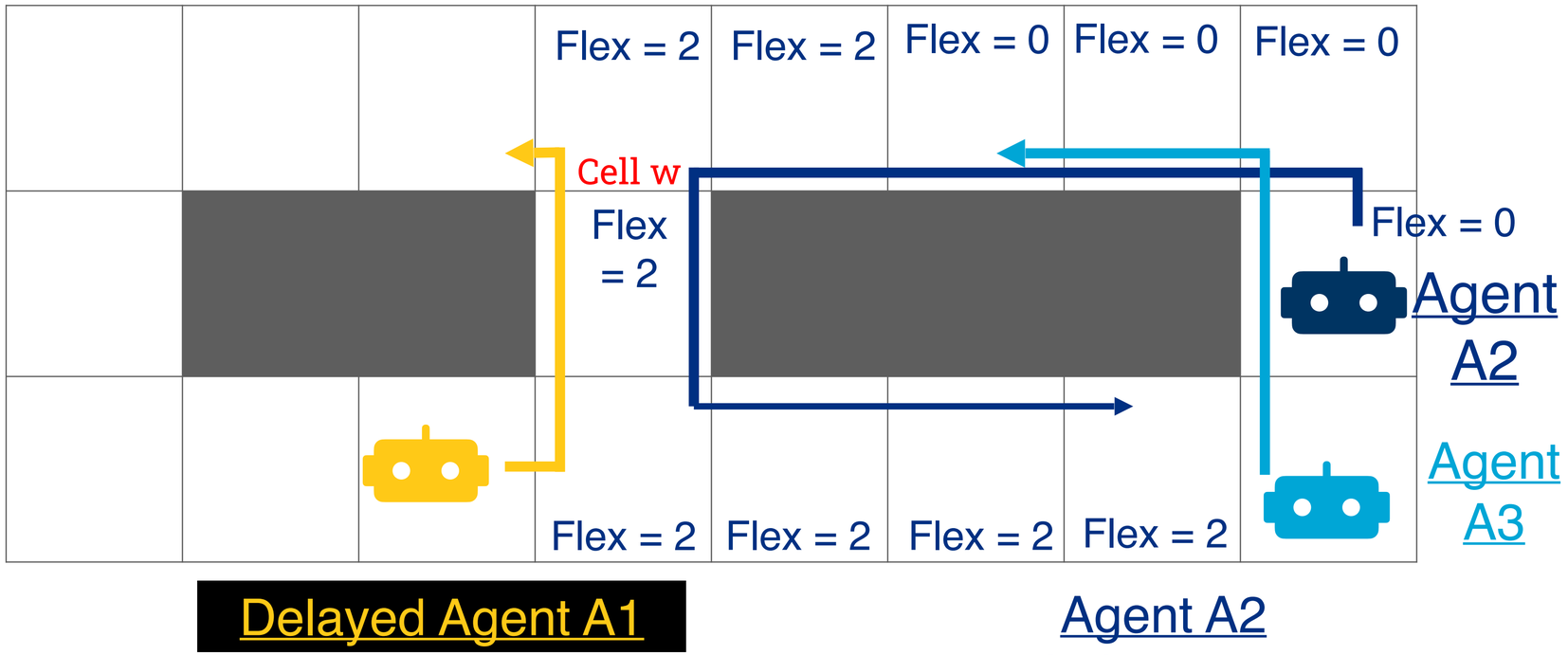


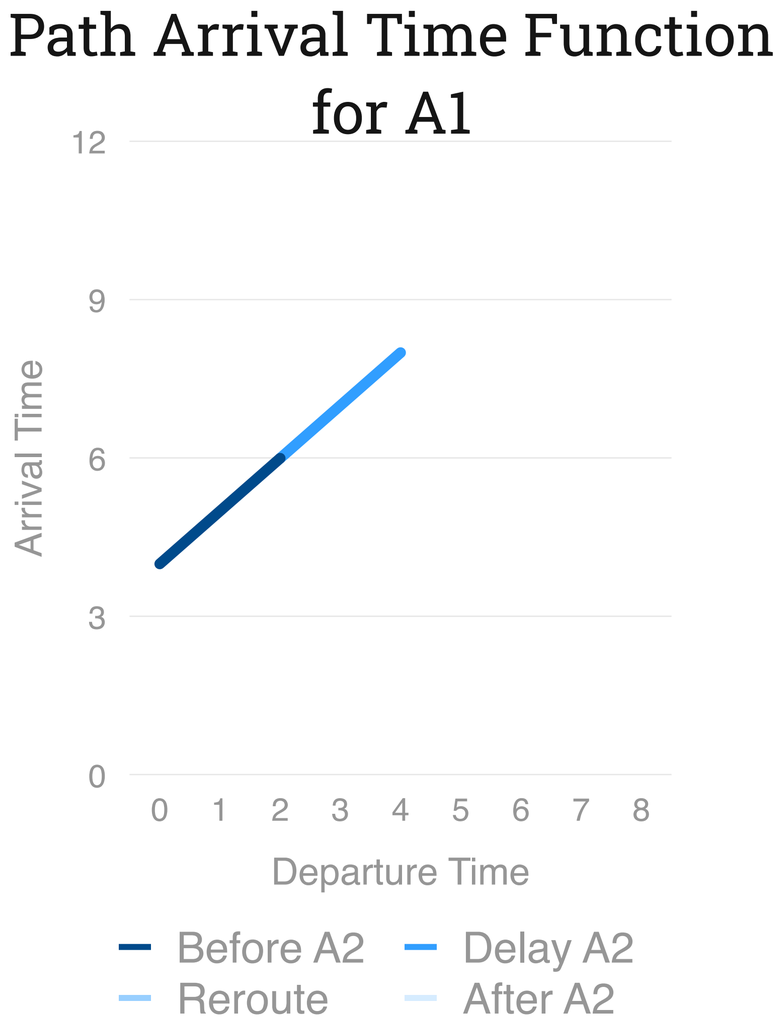
Any-Start-Time Plan



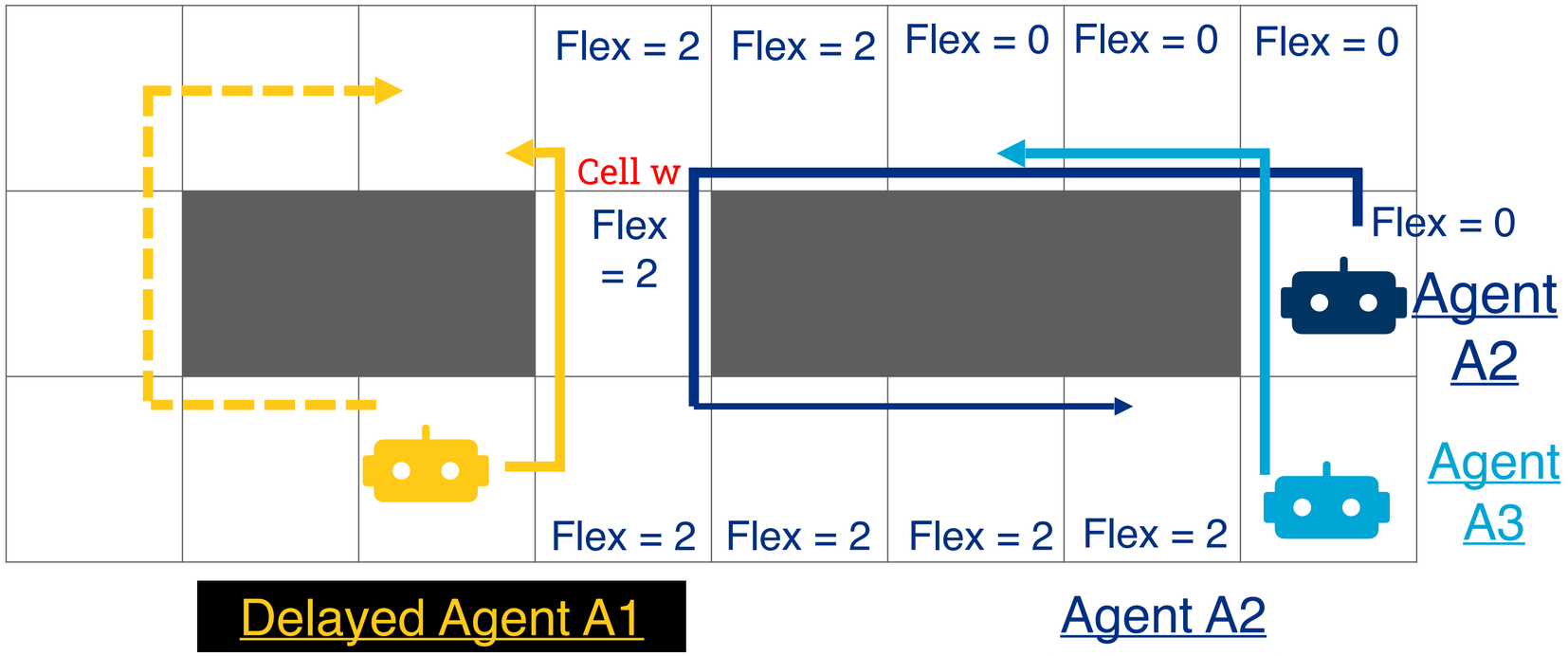


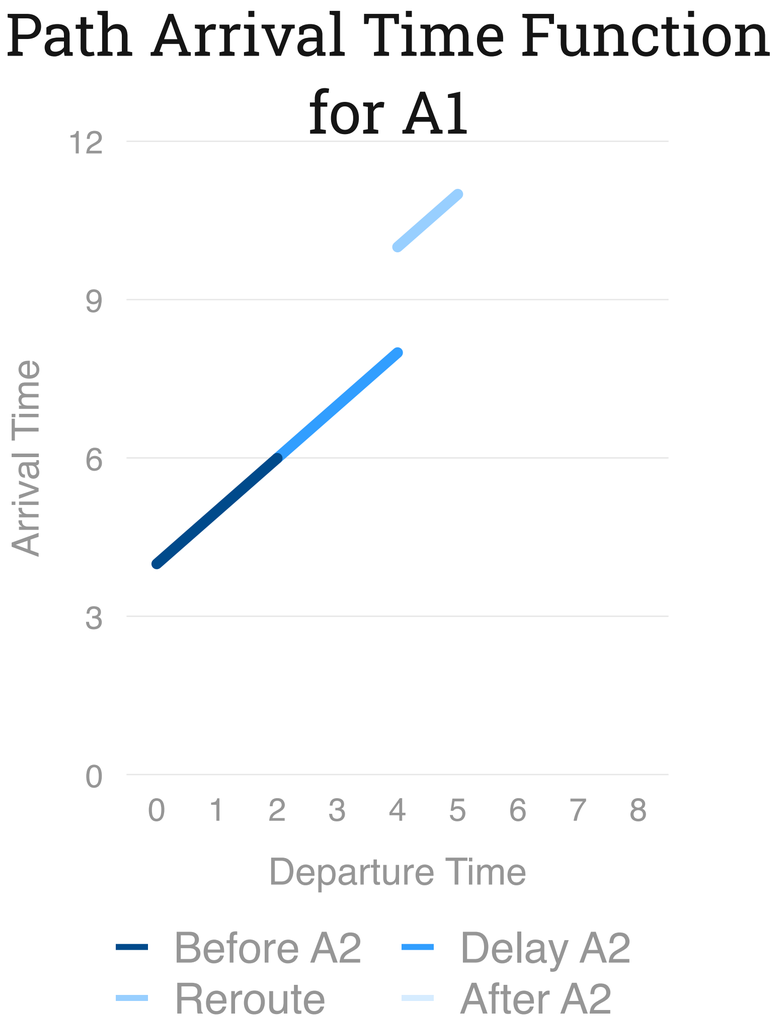
Any-Start-Time Plan



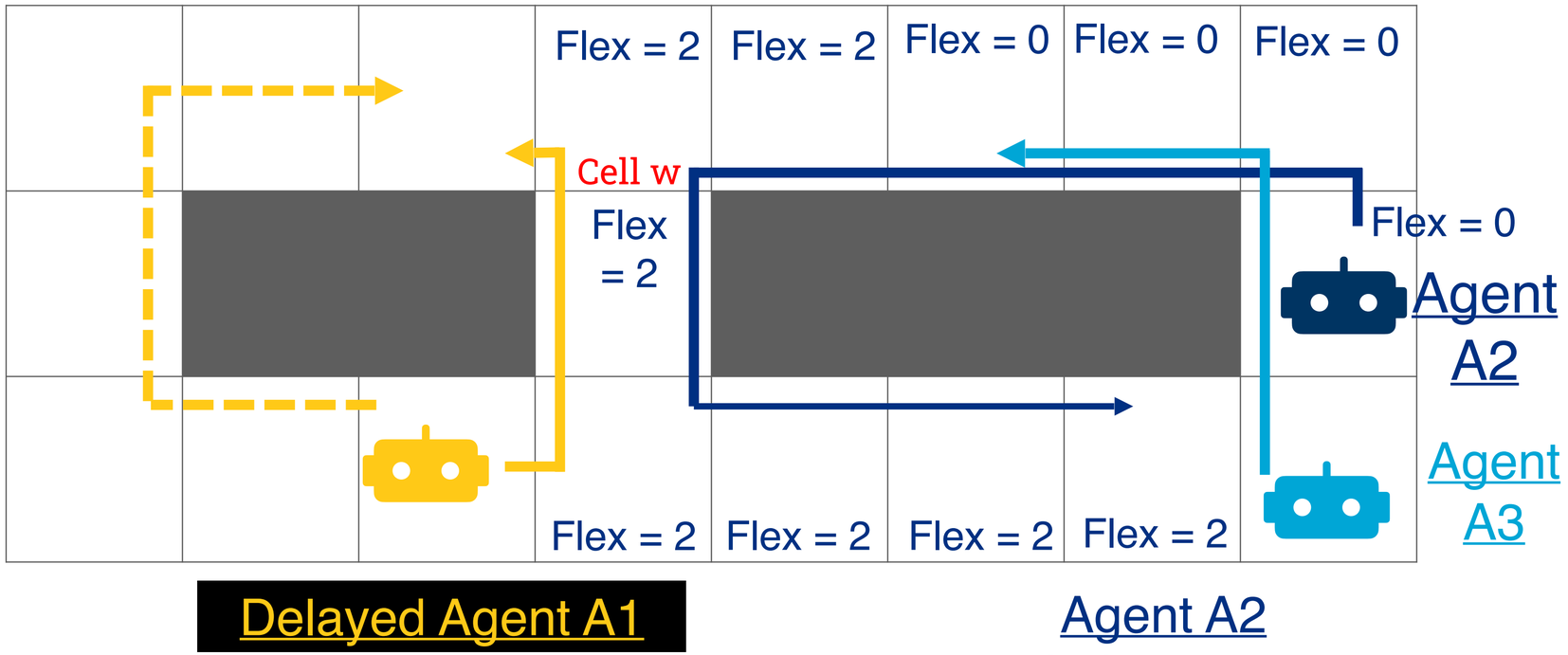


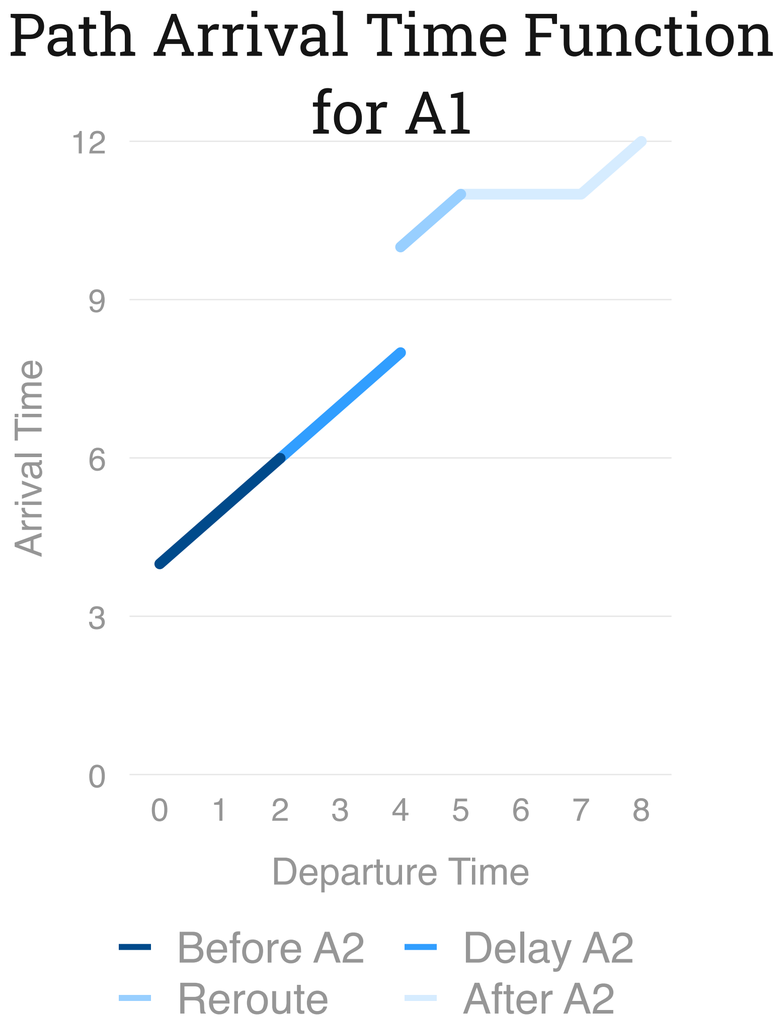
Any-Start-Time Plan





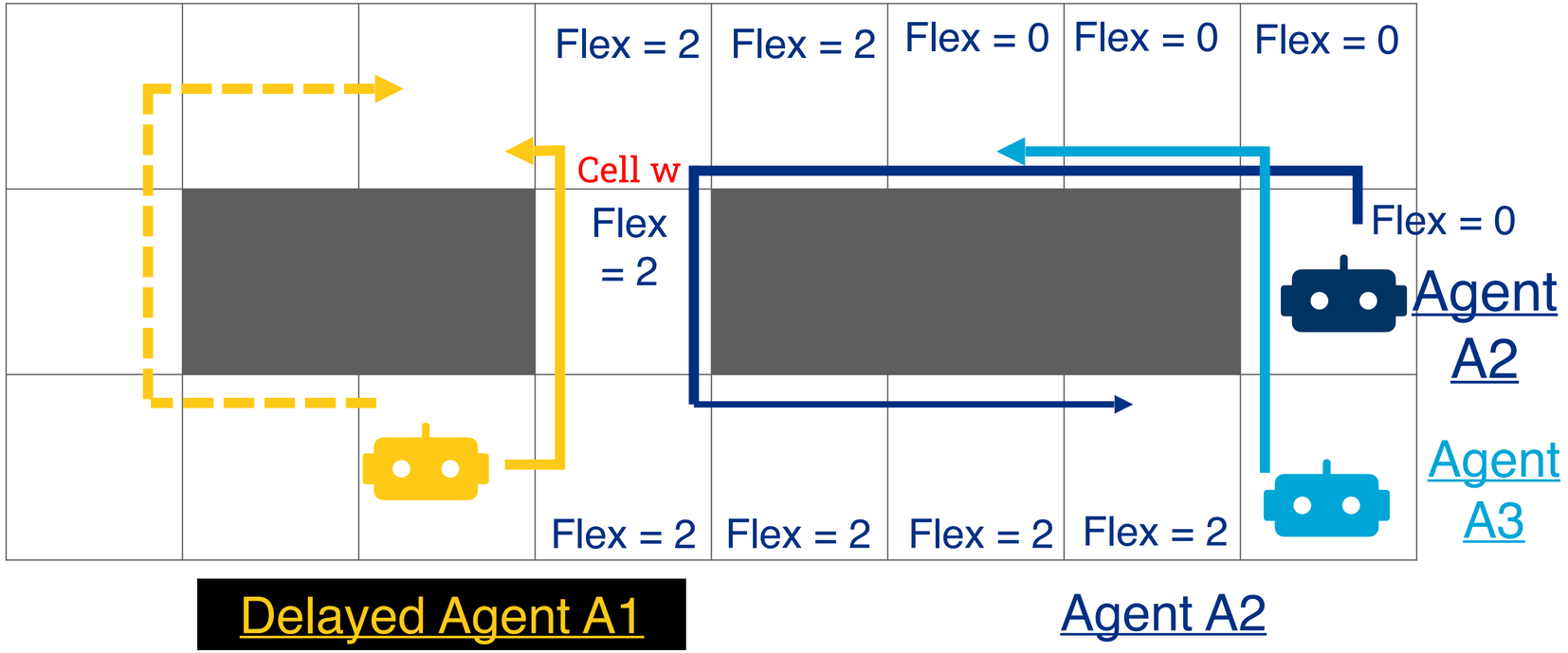
Any-Start-Time Plan

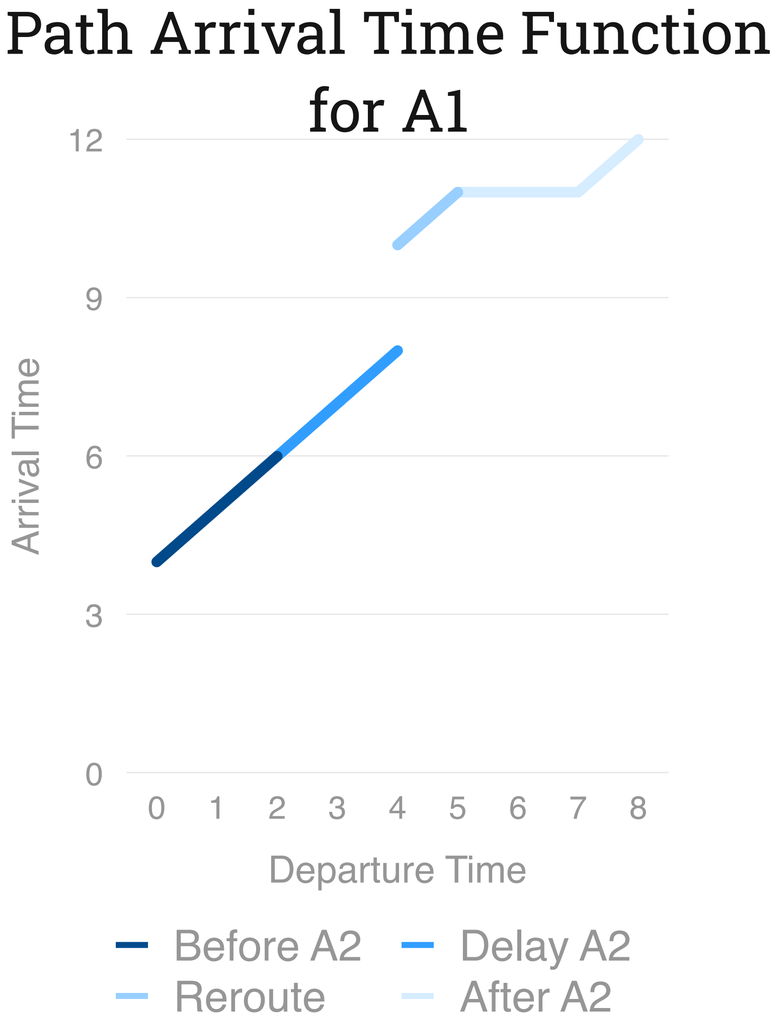




Any-Start-Time Plan

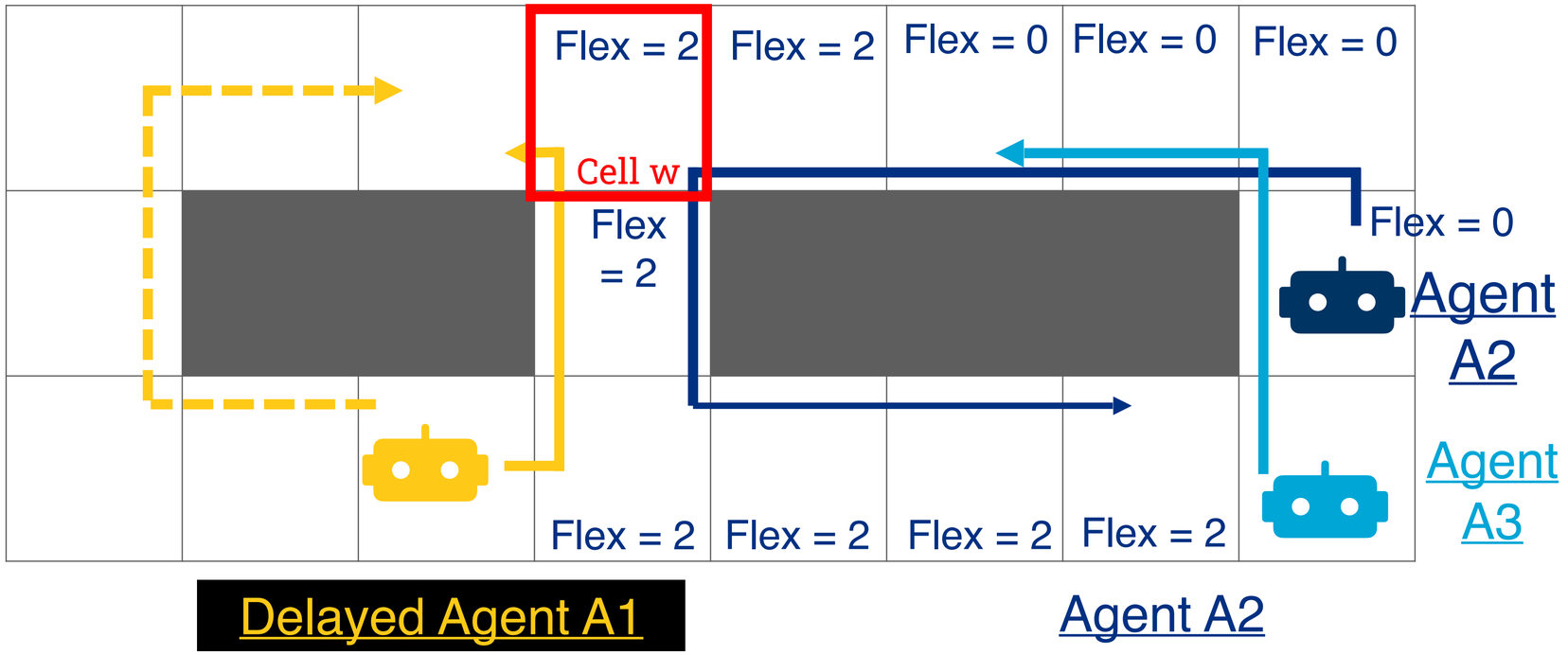
Tipping point is time-location pair where the agent order should change and the other agent should no longer be delayed

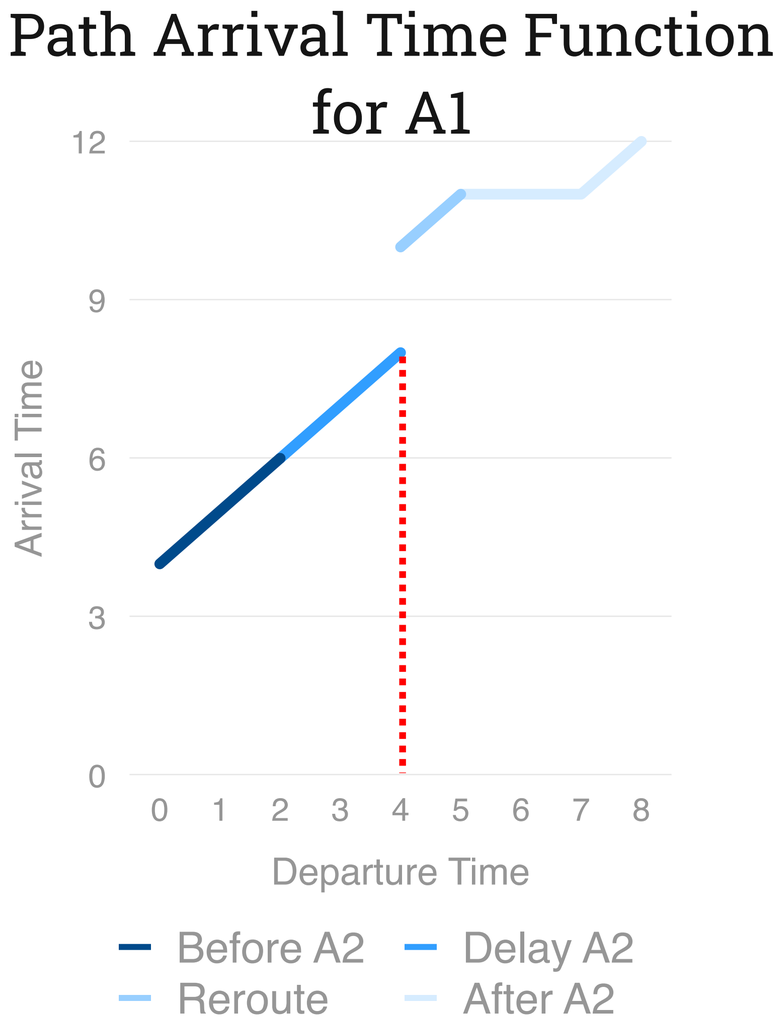




Any-Start-Time Plan

Tipping point is time-location pair where the agent order should change and the other agent should no longer be delayed





Experiments

- MovingAI Benchmark set
  - Maze-128-128-1 with 20 and 50 agents
  - Warehouse-20-40-10-2-1 with 50 agents
- Plans computed using PBS
- Introduce artificial flexibility to show effectiveness

1. Compare overall delay @MAEDeR and FlexSIPP

2. Sequentially delay one random agent

Experiment 1: Any-start-time plans

Experiment 1: Any-start-time plans

Map	k	f
maze	20	0
		3
		5
		8
	50	0
		3
		5
		8
ware house	50	0
		3
		5
		8

Experiment 1: Any-start-time plans

Map	k	f	Delay F / M
maze	20	0	93.4%
		3	93.9%
		5	93.8%
		8	59.9%
	50	0	91.8%
		3	86.1%
		5	93.0%
		8	66.2%
ware house	50	0	100.0%
		3	100.0%
		5	98.1%
		8	102.5%

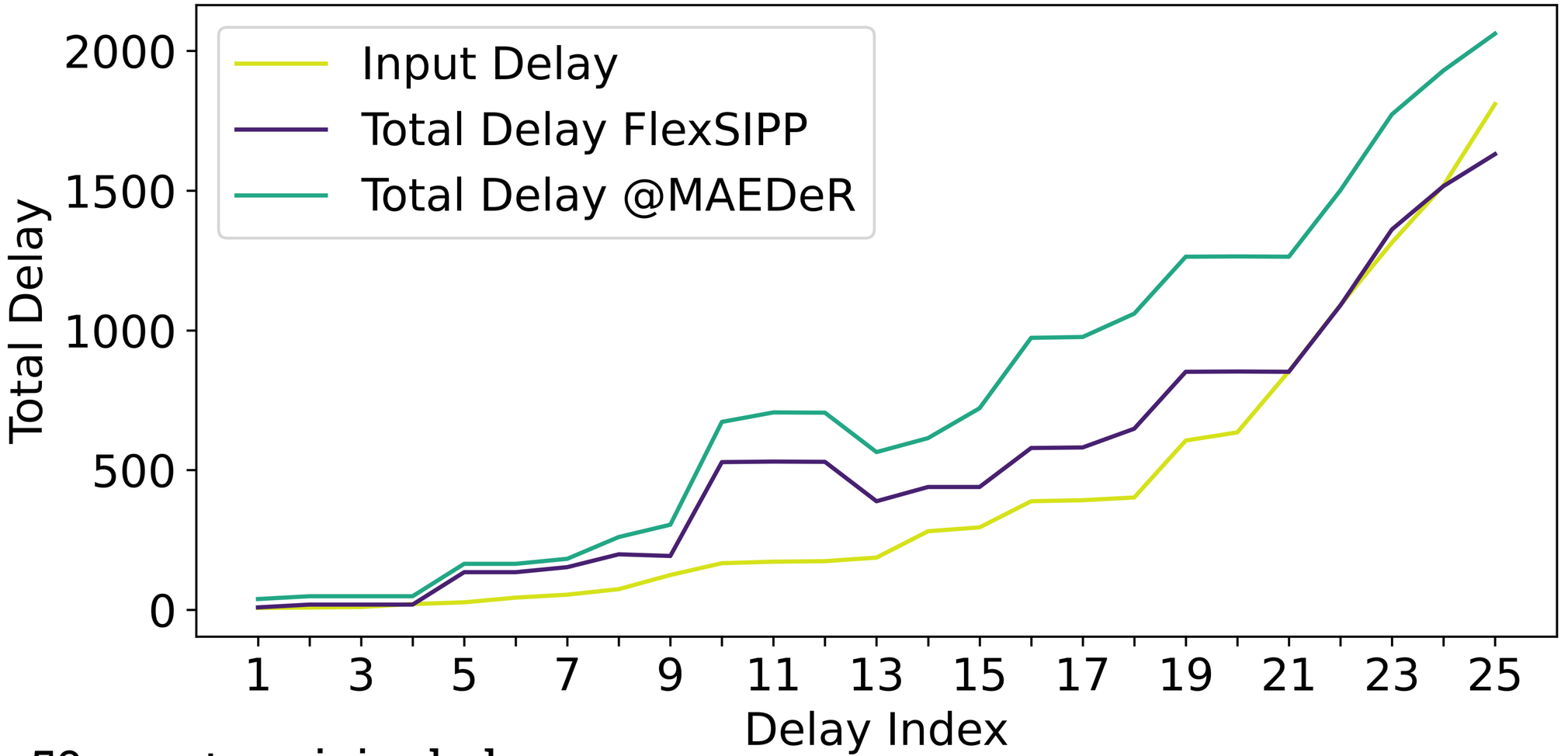
Experiment 1: Any-start-time plans

Map	k	f	Delay F / M	Time (s) F	Time (s) M
maze	20	0	93.4%	0.96	0.55
		3	93.9%	1.51	0.81
		5	93.8%	1.06	0.57
		8	59.9%	2.02	0.80
	50	0	91.8%	2.74	1.51
		3	86.1%	3.78	1.77
		5	93.0%	3.31	1.92
		8	66.2%	3.54	1.87
ware house	50	0	100.0%	3.56	3.26
		3	100.0%	4.89	4.13
		5	98.1%	4.69	4.15
		8	102.5%	3.20	2.52

Experiment 1: Any-start-time plans

Map	k	f	Delay F / M	Time (s) F	Time (s) M	Path Found F	Path Found M
maze	20	0	93.4%	0.96	0.55	66%	64%
		3	93.9%	1.51	0.81	66%	60%
		5	93.8%	1.06	0.57	73%	70%
		8	59.9%	2.02	0.80	64%	57%
	50	0	91.8%	2.74	1.51	68%	64%
		3	86.1%	3.78	1.77	66%	61%
		5	93.0%	3.31	1.92	57%	52%
		8	66.2%	3.54	1.87	70%	61%
ware house	50	0	100.0%	3.56	3.26	70%	70%
		3	100.0%	4.89	4.13	70%	70%
		5	98.1%	4.69	4.15	53%	53%
		8	102.5%	3.20	2.52	63%	63%

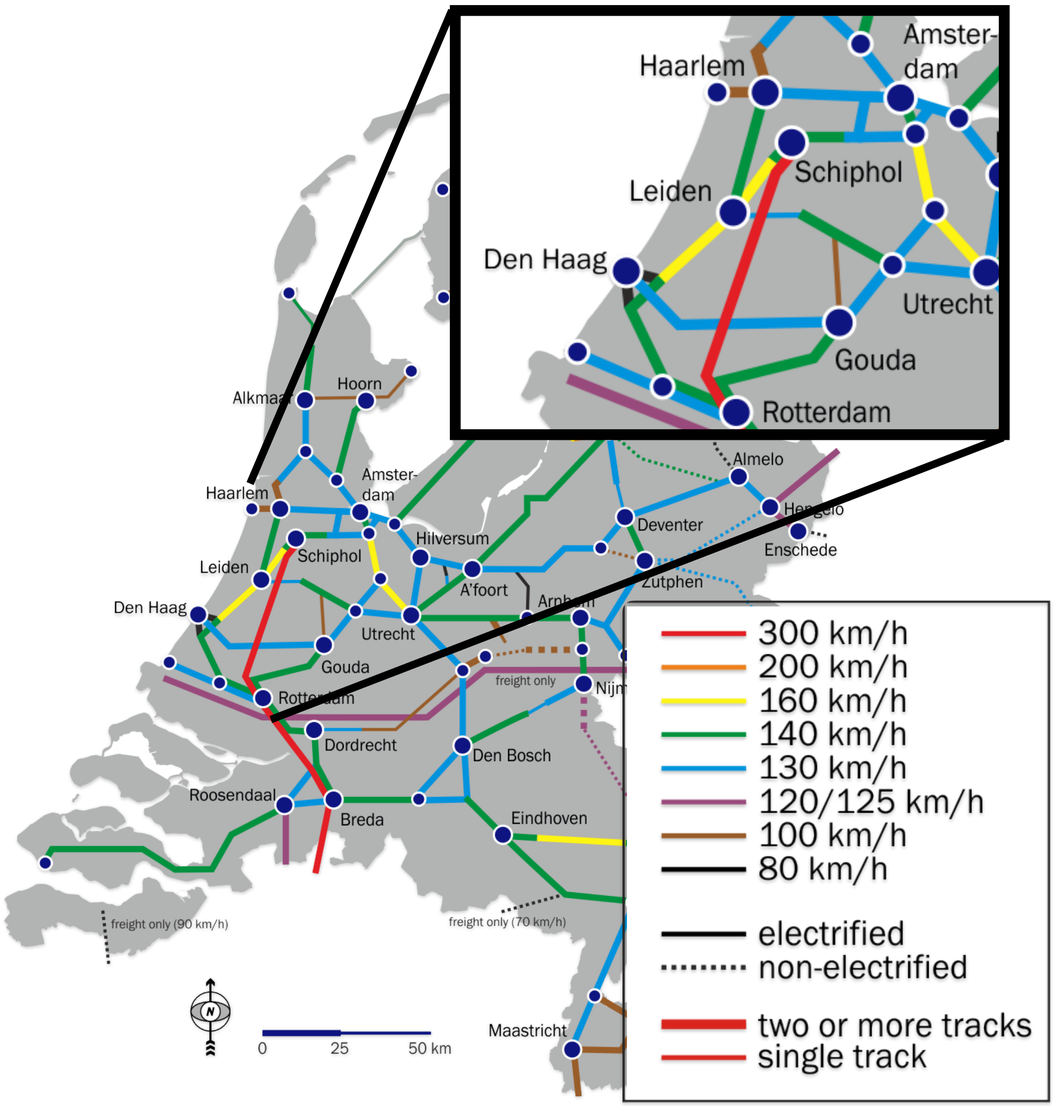
Experiment 2: Sequential Delays



Maze, 50 agents, original plan

Case Study

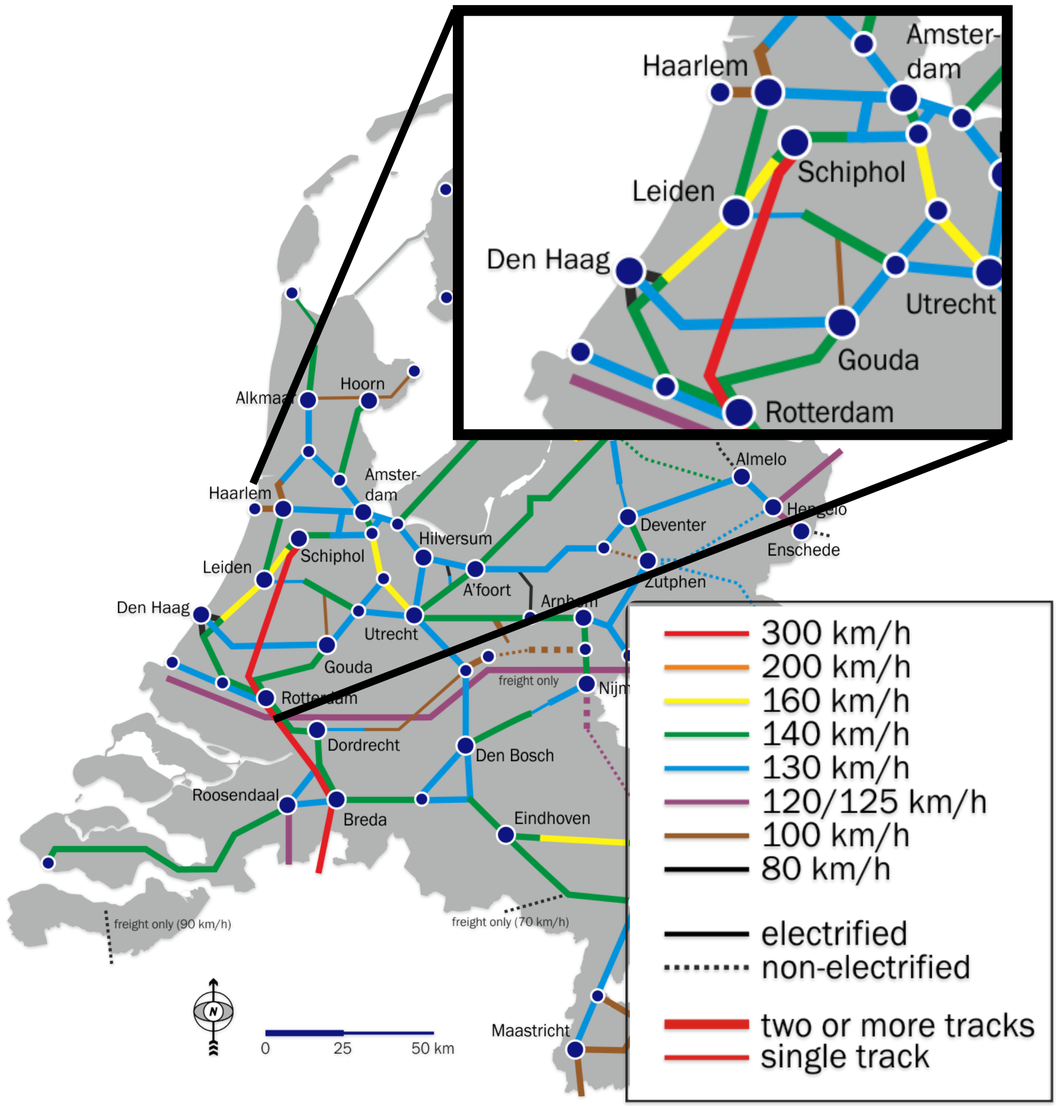
- Real-world timetables from July 2025
  - Introduce new Eurostar train on Schiphol-Leiden-Rotterdam trajectory
- (pre)computation time <5min



Case Study

- Real-world timetables from July 2025
  - Introduce new Eurostar train on Schiphol-Leiden-Rotterdam trajectory
- (pre)computation time <5min

Train	Location	Delay	Tipping point
64	Leiden	00:00	03:59
32	Hoofddorp	02:07	05:00
20	The Hague	01:39	01:16
	Delft	00:00	05:00
51	The Hague	00:00	05:00
16	Leiden	00:00	05:00
11	Hoofddorp	03:42	04:58
70	Schiedam	00:00	05:00
18	Delft	00:00	05:00



Conclusion

Use inherent flexibility in multi-agent plan to instantly recover safely from delay(s)

Future Work

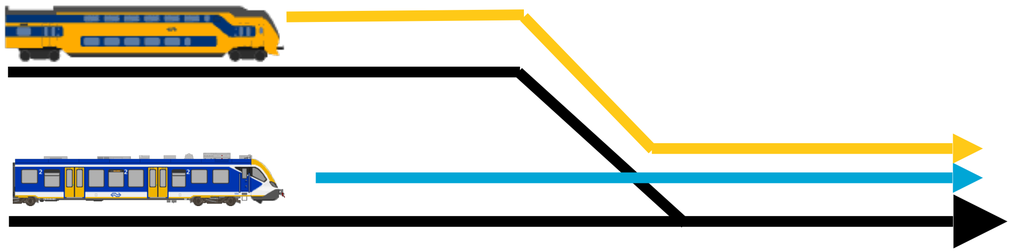
- More extensive experiments with railway simulator to define tipping points
- Apply to your multi-agent environment!

Precomputing Multi-Agent Path Replanning Using Temporal Flexibility

Issa Hanou, Eric Kemmeren, Devin Wild Thomas, Mathijs de Weerdt

- Use inherent temporal flexibility in plan to recover from single delay
- Compute any-start-time plan with flexibility and rerouting options
- Handle sequential delays and minimize the total delay

TU Delft Delft University of Technology



Link to paper