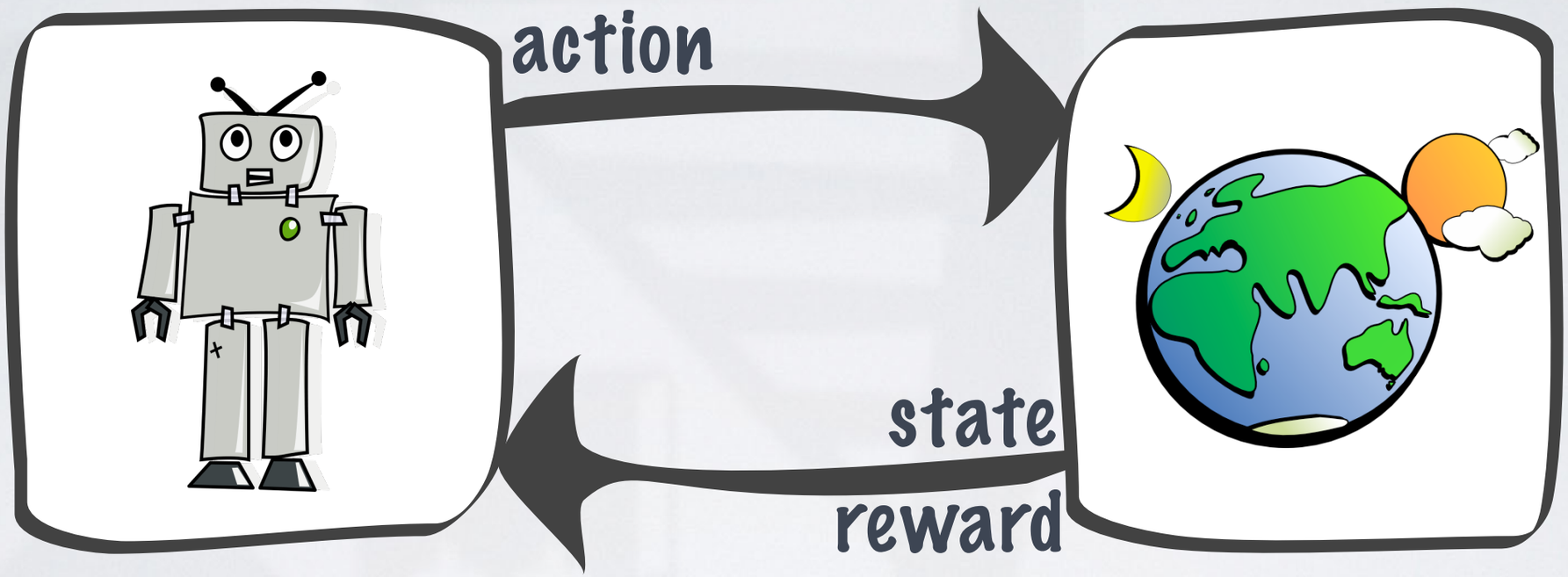
ABSTRACT MDP REWARD SHAPING FOR MULTI-AGENT REINFORCEMENT LEARNING

Kyriakos Efthymiadis, Sam Devlin and Daniel Kudenko,
Department of Computer Science,
The University of York

BACKGROUND MATERIAL

REINFORCEMENT LEARNING



- Traditional RL assumes no prior knowledge
- Including domain knowledge can improve the learning process
- Reward Shaping

REWARD SHAPING

Shaping and SARSA

- SARSA, a popular RL algorithm
  - $Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)]$

- Reward Shaping can provide heuristic knowledge by an additional reward
  - $Q(s,a) \leftarrow Q(s,a) + \alpha[r + F(s,s') + \gamma Q(s',a') - Q(s,a)]$

- Can change the optimal policy when designed poorly

POTENTIAL-BASED REWARD SHAPING

Ng, Russell and Harada (1999)

- Formal definition
  - $F(s, s') = \gamma\Phi(s') - \Phi(s)$

- Guarantees
  - Equivalence to Q-table initialisation
  - Policy invariance (optimal policy unchanged) in single agent

- Can
  - Increase / Decrease time taken to learn optimal policy

MULTI-AGENT POTENTIAL-BASED SHAPING

Devlin and Kudenko (2011)

- Guarantees
  - Equivalence to Q-table initialisation
  - Nash Equilibria not altered

- Can
  - Increase / Decrease time to reach a stable joint policy
  - Change final joint policy

(SEMI-)AUTOMATIC REWARD SHAPING

- Automatic
  - Requires no prior knowledge of the environment
  - Still improves time to learn optimal policy

- Semi-Automatic
  - Encodes prior knowledge into a potential function
  - Performs better than automatic reward shaping if provided knowledge is suitable

ABSTRACT MDP REWARD SHAPING

Marthi (2007)

- Domain knowledge is provided as an abstract MDP
  - Automatic method
    - Samples the environment to form estimations of states, actions and reward function
  - Solved using dynamic programming
  - Resulting *V(s)* shapes the agent

- Modified version to benefit from expert domain knowledge
  - Provide state, action and reward abstractions
  - Need to find right parameter settings
  - Semi-Automatic method

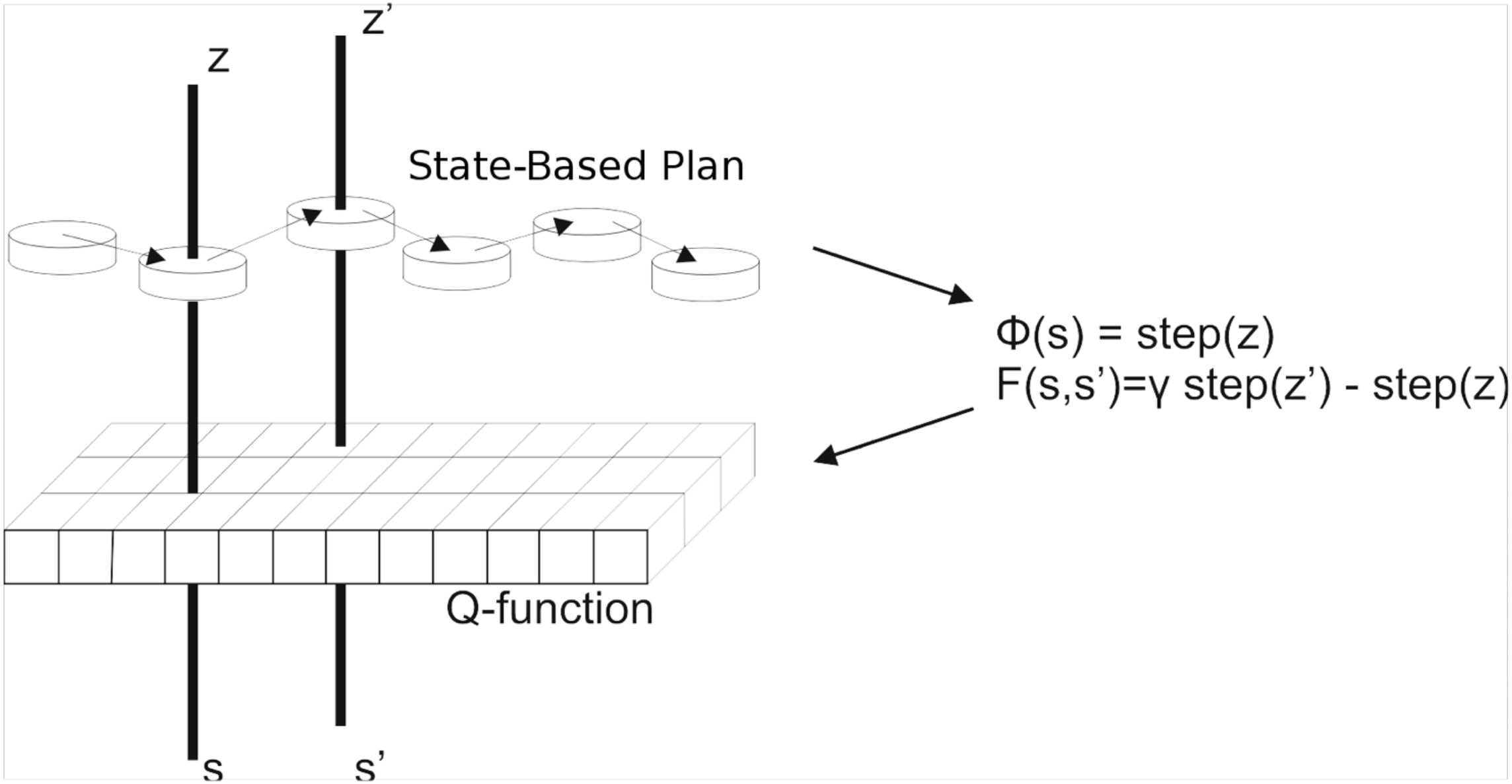
PLAN-BASED REWARD SHAPING

Grzes and Kudenko (2008)

- Domain knowledge is provided as a STRIPS plan
  - Semi-Automatic method
  - Abstraction of the low level environment
    - Start state, goal state, available actions and effects
  - Popular existing tool, familiar to many
  - Easy to generate

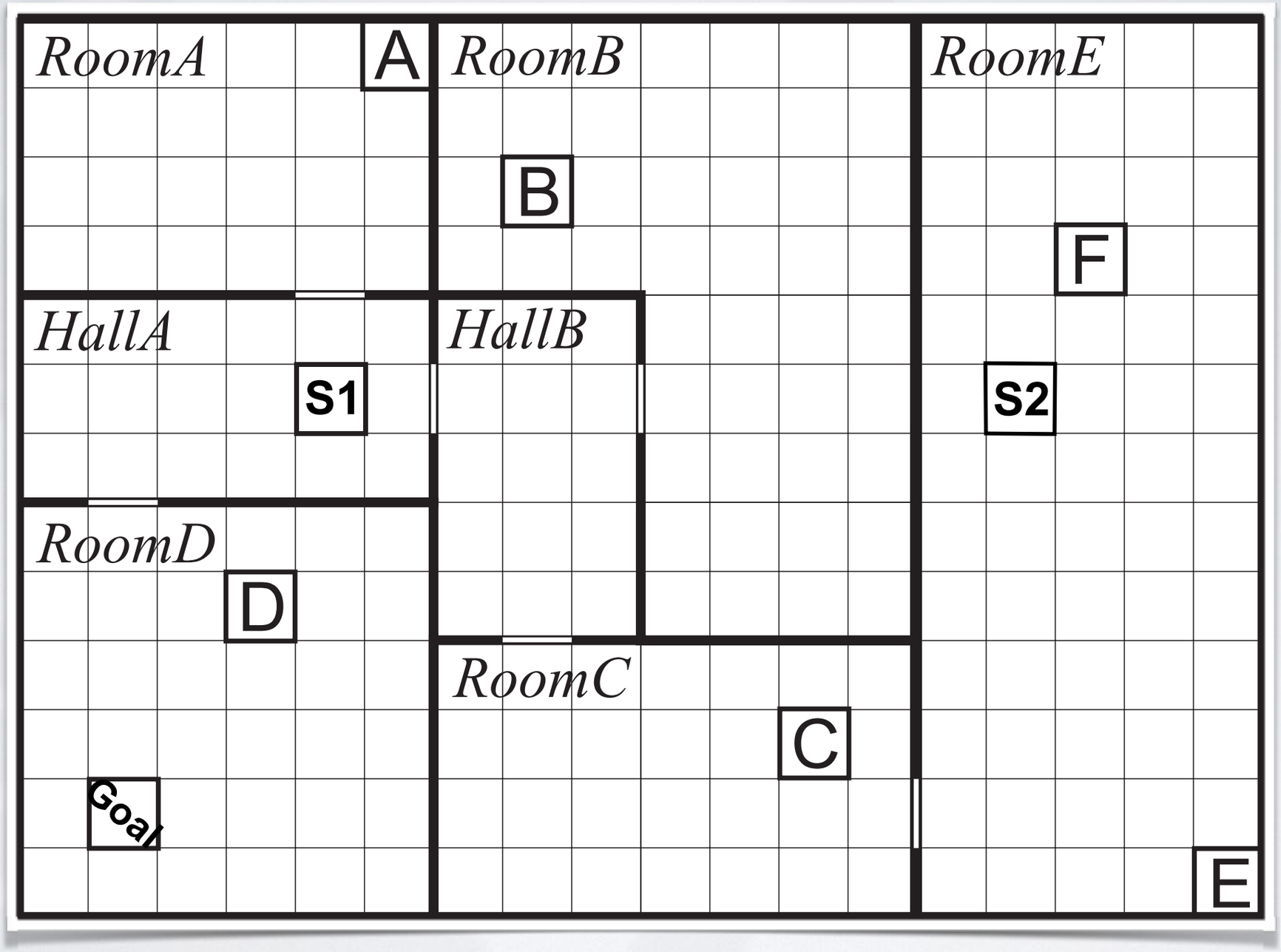
- Generated plan is used to shape the agent
  - Low level states are compared to high level plan states
  - Extra reward is given according to the step in the plan

KNOWLEDGE-BASED RL



EVALUATION DOMAIN

FLAG COLLECTION DOMAIN



Cooperative, Multi-Agent, Deterministic, Discrete Flag Collecting

SHAPING THE AGENT

Transformation

```
move(roomA)

take(flagA)

move(roomB)

take(flagB)

move(roomC)
```

```
in(roomA)

in(roomA)
        have(flagA)

in(roomB)
        have(flagA)

in(roomB)
    have(flagA,flagB)

in(roomC)
    have(flagA,flagB)
```

SHAPING THE AGENT

During Simulation

$$F(s, s') = \gamma\Phi(s') - \Phi(s)$$

ABS

```
35 in(roomA)

40 in(roomA)
       have(flagA)

59 in(roomB)
       have(flagA)

70 in(roomB)
    have(flagA,flagB)

97 in(roomC)
    have(flagA,flagB)
```

PB

```
1 in(roomA)

2 in(roomA)
       have(flagA)

3 in(roomB)
       have(flagA)

4 in(roomB)
    have(flagA,flagB)

5 in(roomC)
    have(flagA,flagB)
```

SHAPING FOR MULTIPLE AGENTS

- Joint
  - Centralised planning
  - Agents share goals and capabilities
  - Provides coordination knowledge to agents

- Individual
  - Decentralised planning
  - Agents not required to share information
  - Efficient use of available resources
  - Smaller state-action space
  - Conflicting goals

RESULTS

RESULTS

Setup

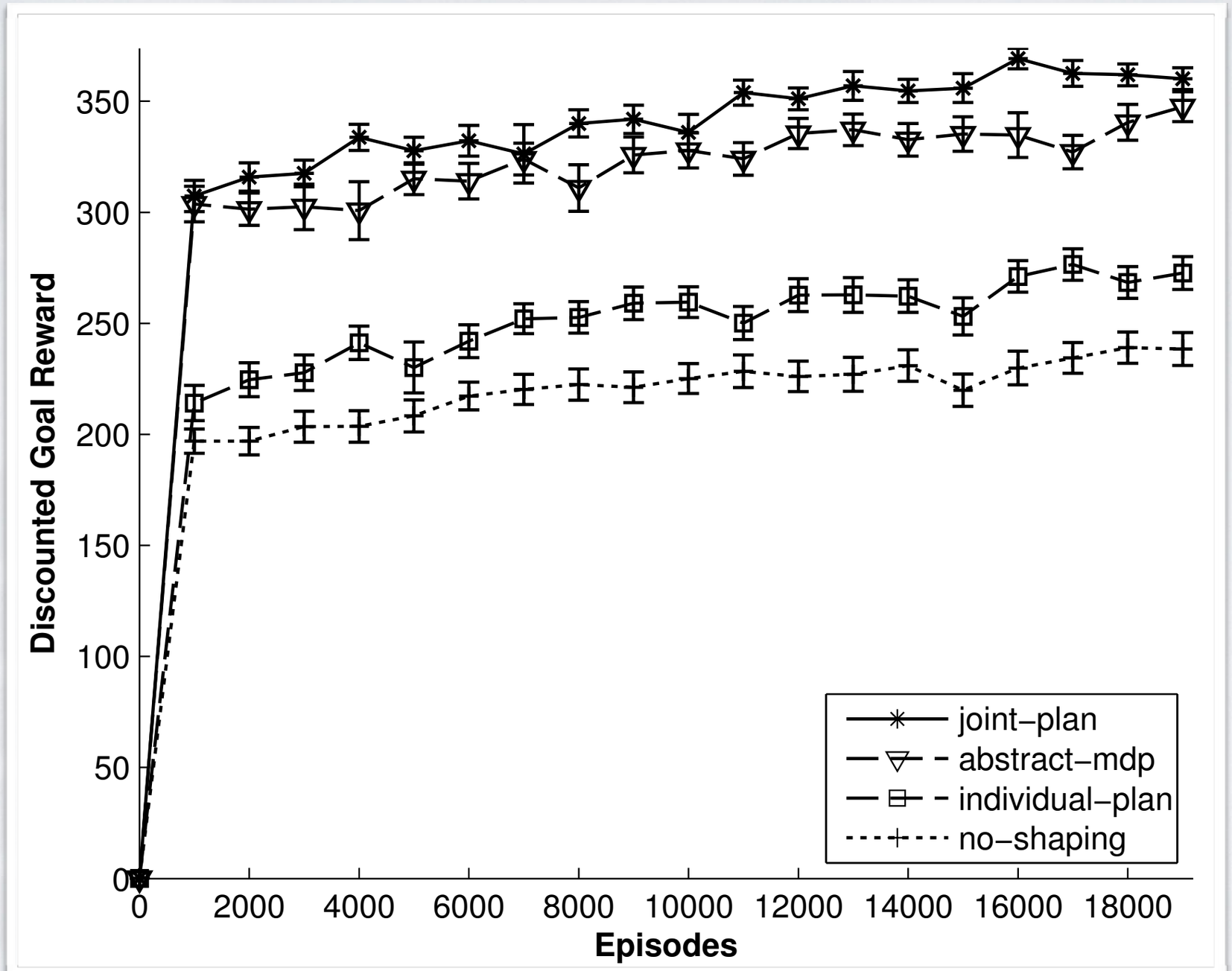
- Variation of the flag collection domain
  - 6 flags and 7 rooms, 12 flags and 7 rooms, 12 flags and 12 rooms

- Use of a centralised agent as an upper bound
  - joint-plan-based

- Compare abstract MDP vs plan-based agents receiving decentralised shaping

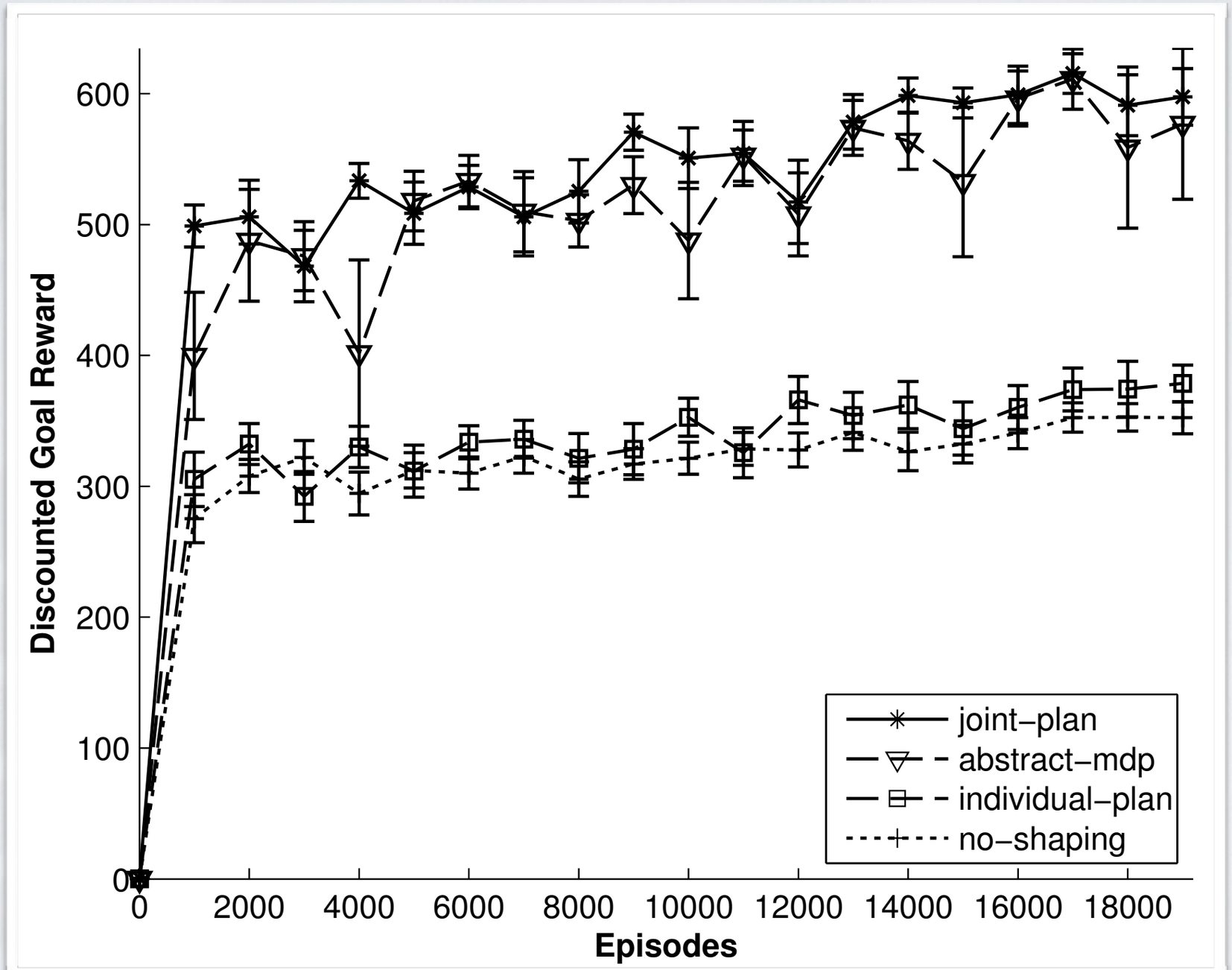
RESULTS

6 flags 7 rooms



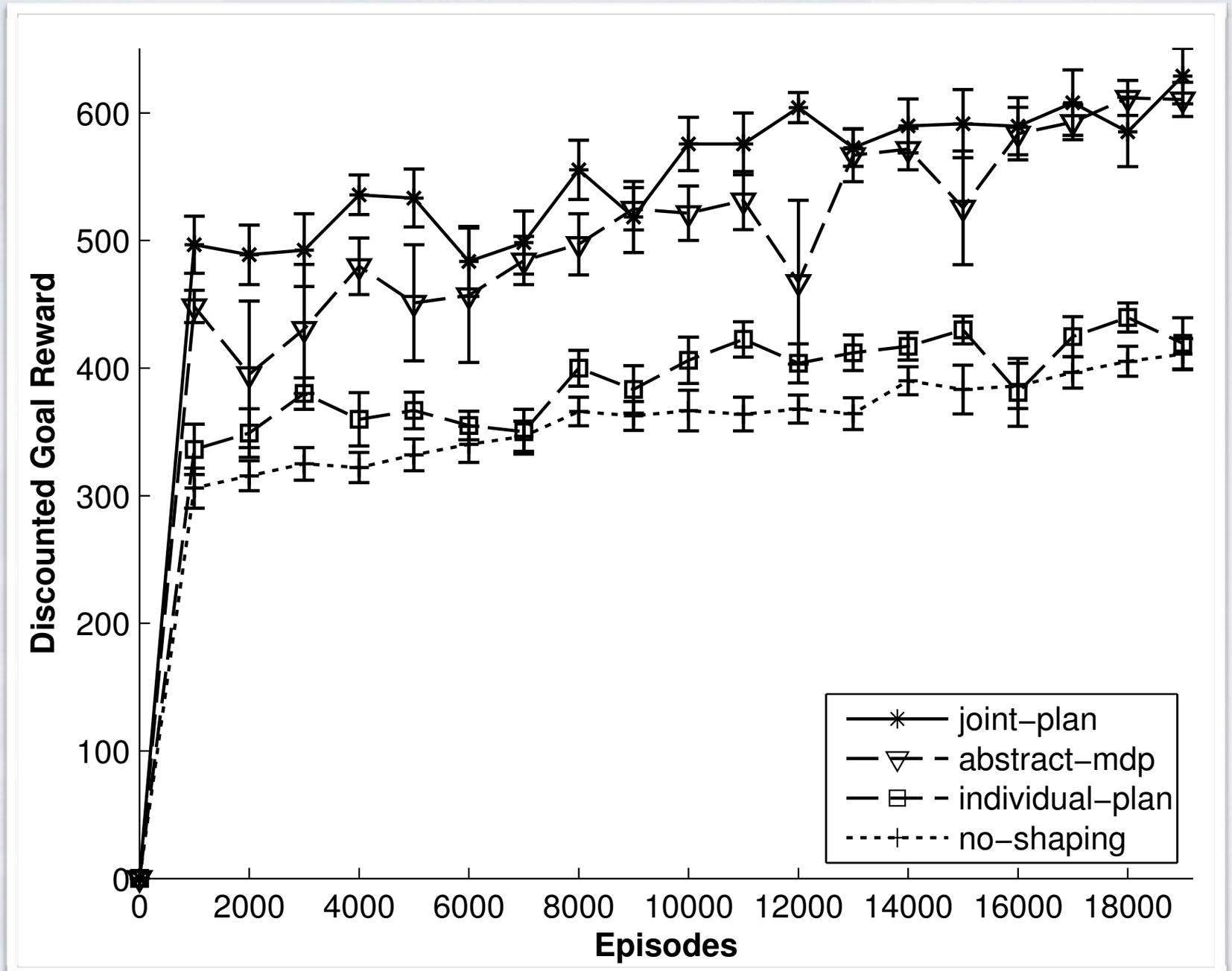
RESULTS

12 flags 7 rooms



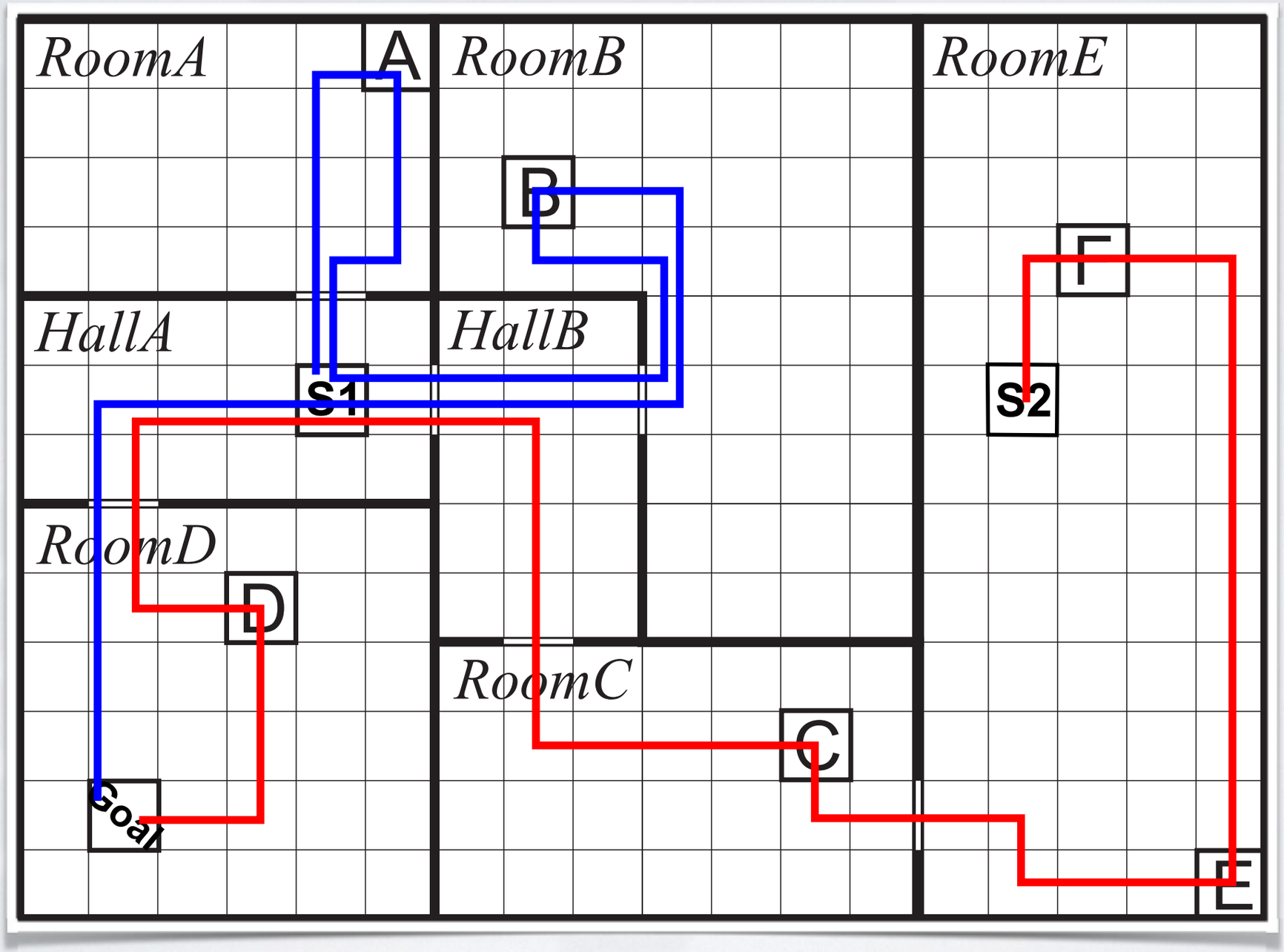
RESULTS

12 flags 12 rooms



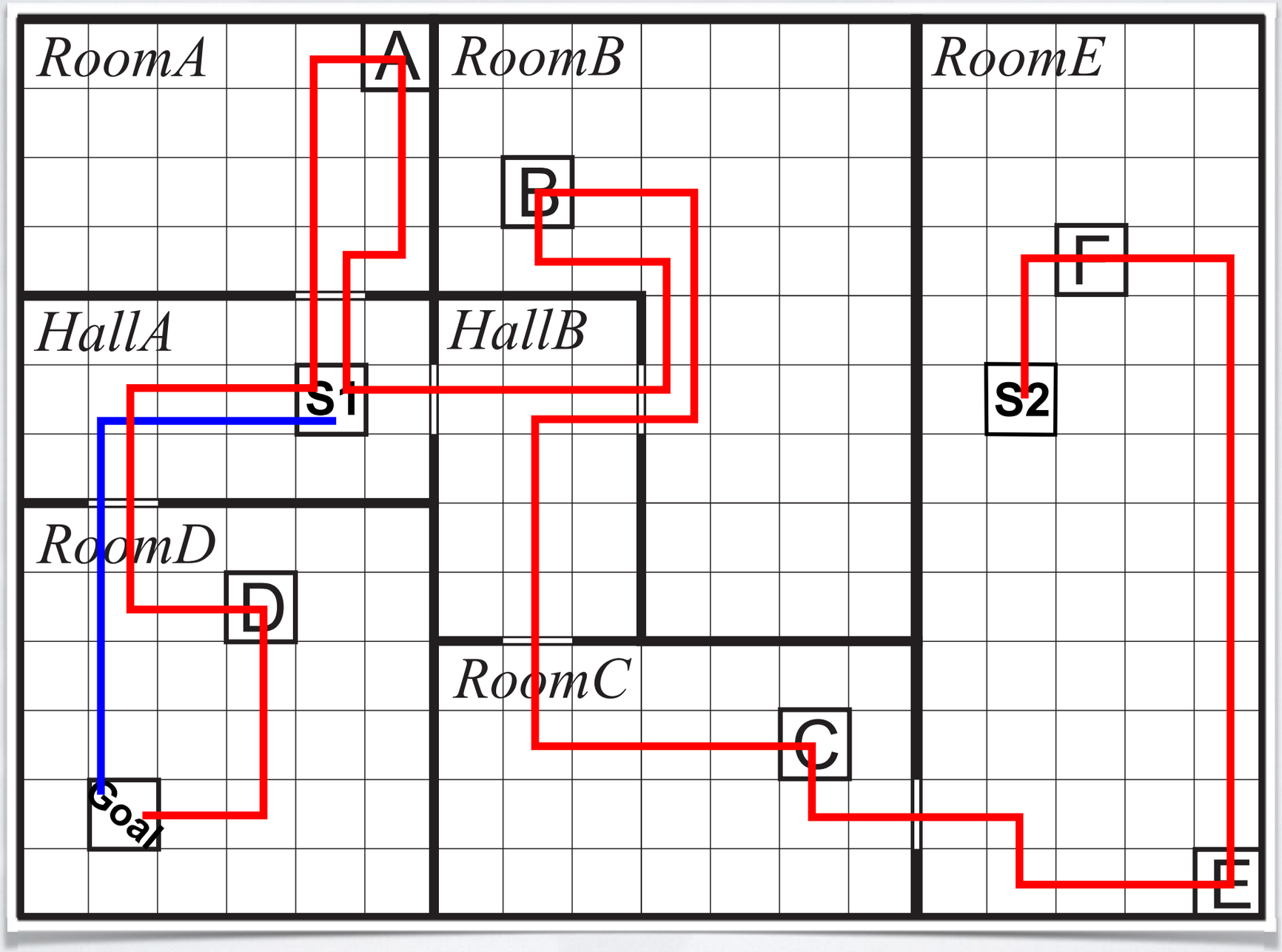
JOINT-PLAN / ABSTRACT MDP

Sample Paths



INDIVIDUAL PLAN-BASED

Sample Paths



PLAN-BASED REVISITED

During Simulation

PB

```
1 in(roomA)

2 in(roomA)
      have(flagA)

3 in(roomB)
      have(flagA)

4 in(roomB)
    have(flagA,flagB)

5 in(roomC)
    have(flagA,flagB)
```

ABSTRACT MDP REVISITED

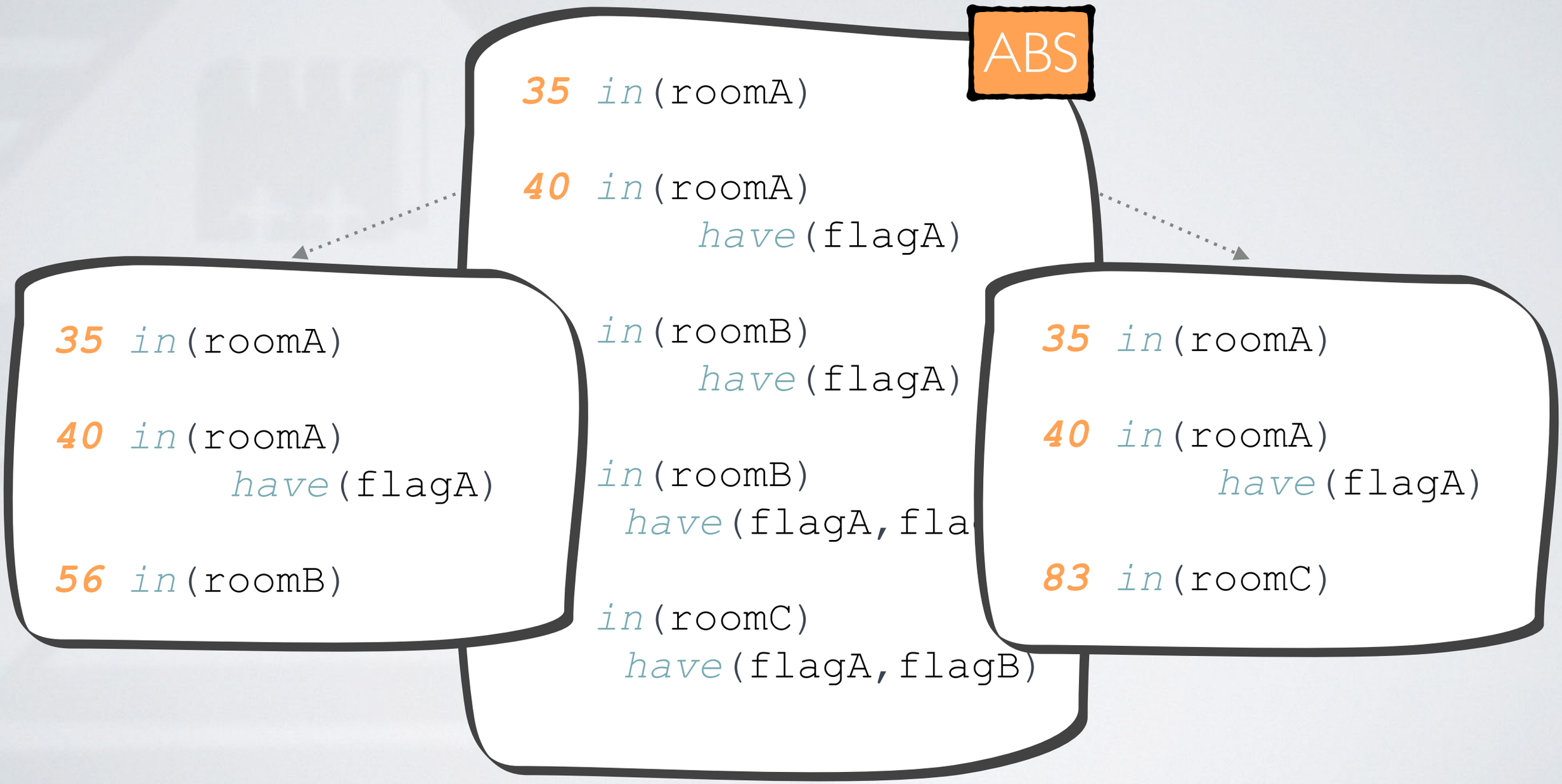
During Simulation

ABS

```
35 in(roomA)

40 in(roomA)
       have(flagA)

59 in(roomB)
       have(flagA)

70 in(roomB)
    have(flagA,flagB)

97 in(roomC)
    have(flagA,flagB)
```

ABSTRACT MDP REVISITED

During Simulation



CONCLUSION

CONCLUSION

- Proposed abstract MDP shaping for multi-agent environments
- Abstract MDP agents can cooperate despite given decentralised reward shaping
- Eliminates the impact of conflicting goals
- Can be used both as a shaping method (speed up the learning process) as well as for conflict resolution (learn better policies)
- Can be tricky to formulate and get the parameters "right"
- Cost of solving the MDP can be high in very large environments

Thank you for listening.
Any questions?