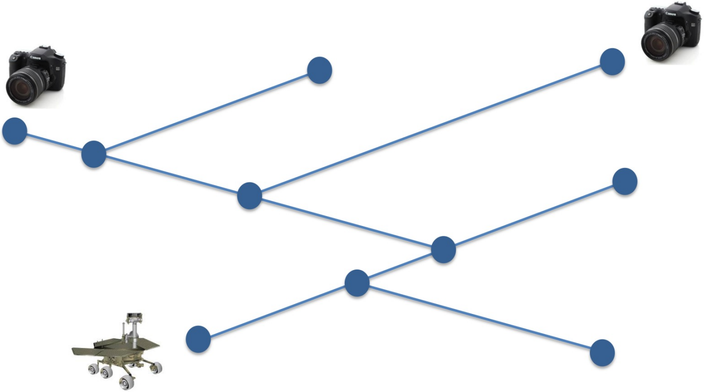
Anticipatory Search as Partial Satisfaction Planning with State Dependent Costs

Daniel Borrajo   Raquel Fuentetaja   Tomás de la Rosa

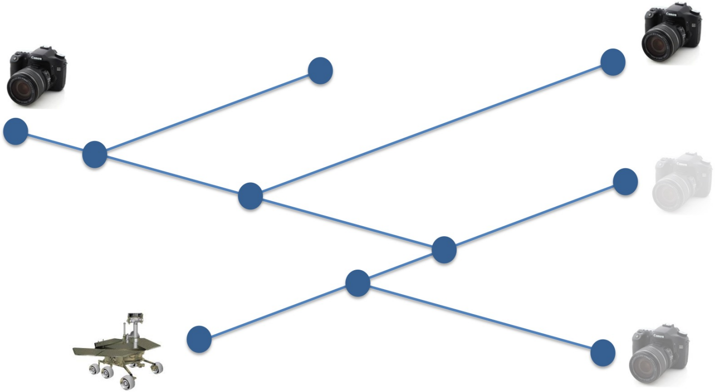
Universidad Carlos III de Madrid

Workshop on Goal Reasoning. IJCAI'16

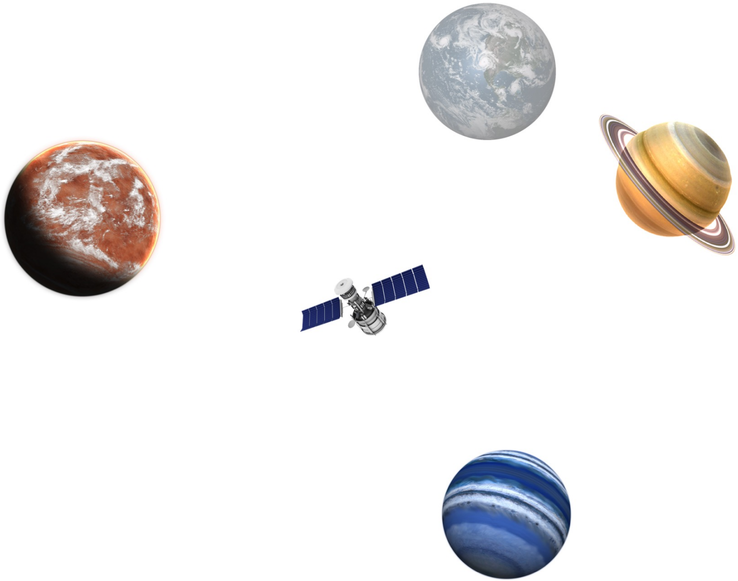
Motivation



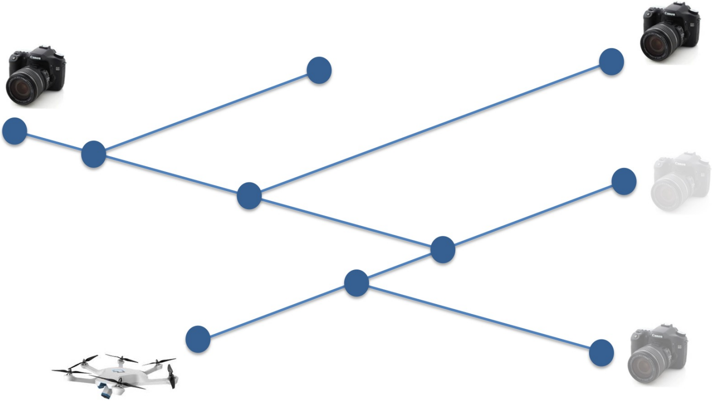
Motivation



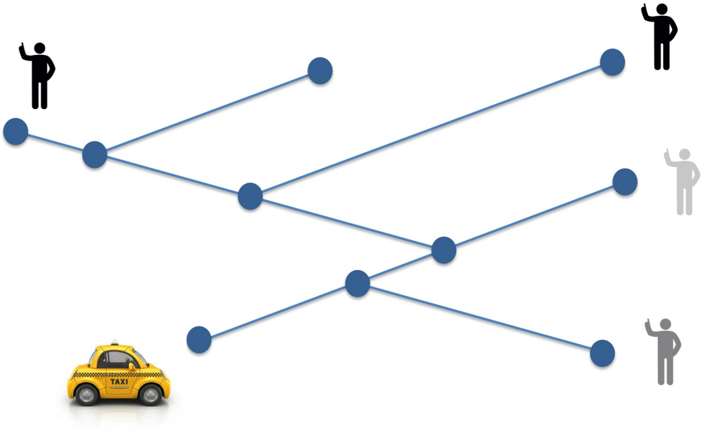
Motivation



Motivation



Motivation



On-line Continual Planning Problem (OCPP)

MDP $\langle S, A, T, C \rangle$

- Set of states, S. $s \in S$, $s = (w, G) \in W \times \mathcal{G}$
- Set of actions, A
- Transition function, $T : S \times A \times S \rightarrow [0, 1]$
- Cost function, $C : S \times A \times S \rightarrow \mathbb{R}_0^+$

[Burns *et al.*, 2012]

On-line Continual Planning Problem (OCPP)

Assumptions

- Actions are
  - deterministic on states
  - stochastic on goals

On-line Continual Planning Problem (OCPP)

Assumptions

- Actions are
  - deterministic on states
  - stochastic on goals
- Goals set monotonically increases

On-line Continual Planning Problem (OCPP)

Assumptions

- Actions are
  - deterministic on states
  - stochastic on goals
- Goals set monotonically increases
- Goals are independent of states
- Goals are independent of each other

On-line Continual Planning Problem (OCPP)

Assumptions

- Actions are
  - deterministic on states
  - stochastic on goals
- Goals set monotonically increases
- Goals are independent of states
- Goals are independent of each other
- Reasoning on a finite horizon H

On-line Continual Planning Problem (OCPP)

Assumptions

- Actions are
  - deterministic on states
  - stochastic on goals
- Goals set monotonically increases
- Goals are independent of states
- Goals are independent of each other
- Reasoning on a finite horizon H

Objective

Minimize the sum of

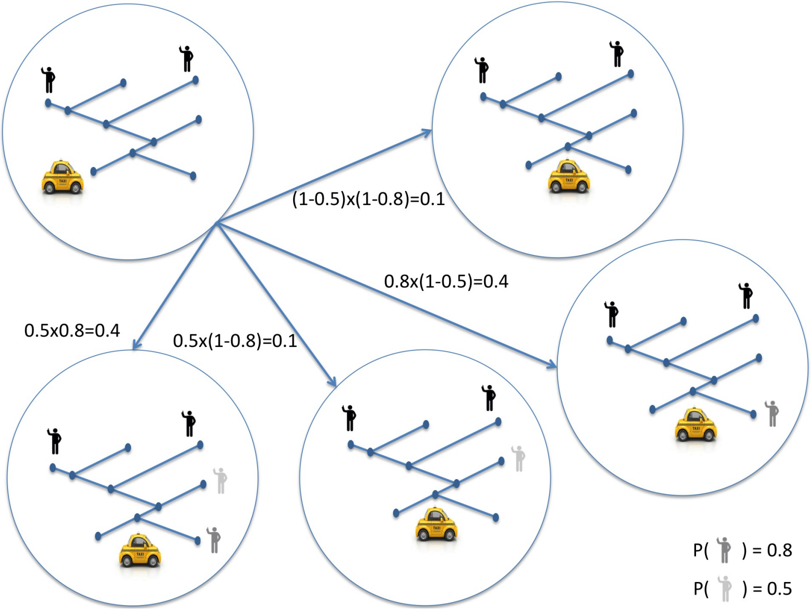
- penalties for unachieved goals and
- actions costs

Transition function

$$T(s,a,s') = \begin{cases} 0 & \text{if } a \text{ not applicable in } s \\ P(G' \mid G) & \textit{otherwise} \end{cases}$$

$$P(G'|G) = \begin{cases} 0 & \text{if } G \not\subseteq G' \\ 1 & \text{if } G = G' = \mathcal{G} \\ \Big[\prod_{g \in G' - G} P(g)\Big] \times \Big[\prod_{g \in \mathcal{G} - G'} (1 - P(g))\Big] & \text{otherwise} \end{cases}$$

Transition function



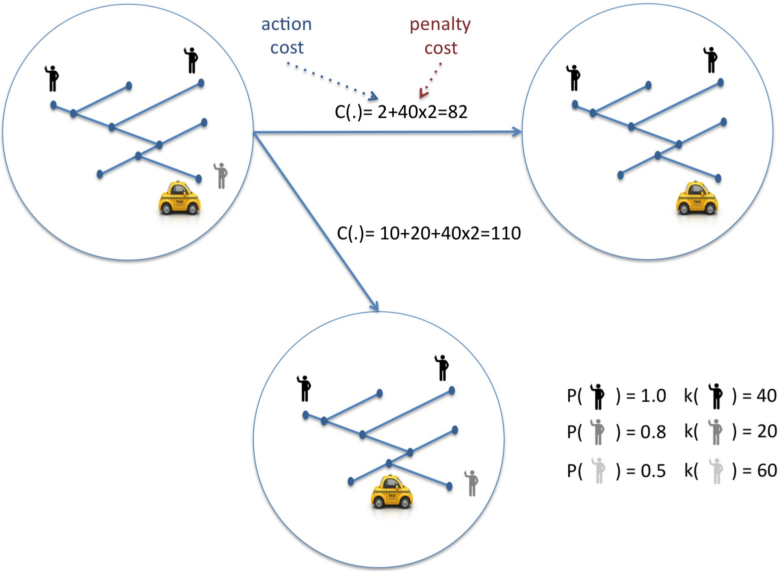
Cost function

$$C(a, s') = C(a, (w', G')) = cost(a) + penalty(w', G')$$

$$penalty(w', G') = \sum_{g' \in G'} penalty(w', g')$$

$$penalty(w', g') = \begin{cases} k_{g'} & \text{if } g' \notin w' \\ 0 & \text{otherwise} \end{cases}$$

Cost function



From OCPP to Automated Planning

- Efficient problem solving
  $\Longrightarrow$ OCPP (MDP) as Automated Planning (AP)

From OCPP to Automated Planning

- Efficient problem solving
  $\Longrightarrow$ OCPP (MDP) as Automated Planning (AP)
- Domain-independent problem solving
  vs. Hindsight Optimization (HO)

From OCPP to Automated Planning

- Efficient problem solving
  $\Longrightarrow$ OCPP (MDP) as Automated Planning (AP)
- Domain-independent problem solving
  vs. Hindsight Optimization (HO)
- Reasoning on a finite horizon H
  $\Longrightarrow$ AP use of a search horizon
  $\Longrightarrow$ AP use of soft goals

From OCPP to Automated Planning

- Efficient problem solving
  $\Longrightarrow$ OCPP (MDP) as Automated Planning (AP)
- Domain-independent problem solving
  vs. Hindsight Optimization (HO)
- Reasoning on a finite horizon H
  $\Longrightarrow$ AP use of a search horizon
  $\Longrightarrow$ AP use of soft goals
- Objective: minimize sum of action costs and penalties
  $\Longrightarrow$ AP use of state-dependent action costs

From OCPP to Automated Planning

- Efficient problem solving
  $\Longrightarrow$ OCPP (MDP) as Automated Planning (AP)
- Domain-independent problem solving
  vs. Hindsight Optimization (HO)
- Reasoning on a finite horizon H
  $\Longrightarrow$ AP use of a search horizon
  $\Longrightarrow$ AP use of soft goals
- Objective: minimize sum of action costs and penalties
  $\Longrightarrow$ AP use of state-dependent action costs

Partial Satisfaction Planning with Horizon and State-Dependent costs (PSP-HSD)

Partial Satisfaction Planning with Horizon and State-Dependent costs

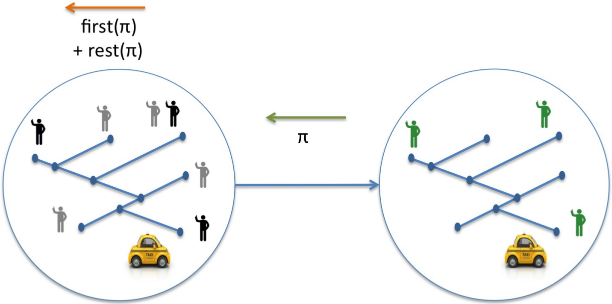
PSP-HSD, $\Pi = (\mathcal{F}, \mathcal{A}, \mathcal{I}, SG, \mathcal{C}, H)$

- $\mathcal{F}$, finite set of fluents
- $\mathcal{A}$, finite set of actions
- $\mathcal{I} \subseteq \mathcal{F}$, initial state
- $SG \subseteq \mathcal{F}$, set of soft goals
- $\mathcal{C} : \mathcal{A} \times \mathcal{W} \to \mathbb{R}_0^+$, state-dependent action cost function
- $H \in \mathbb{N}_0$, finite horizon

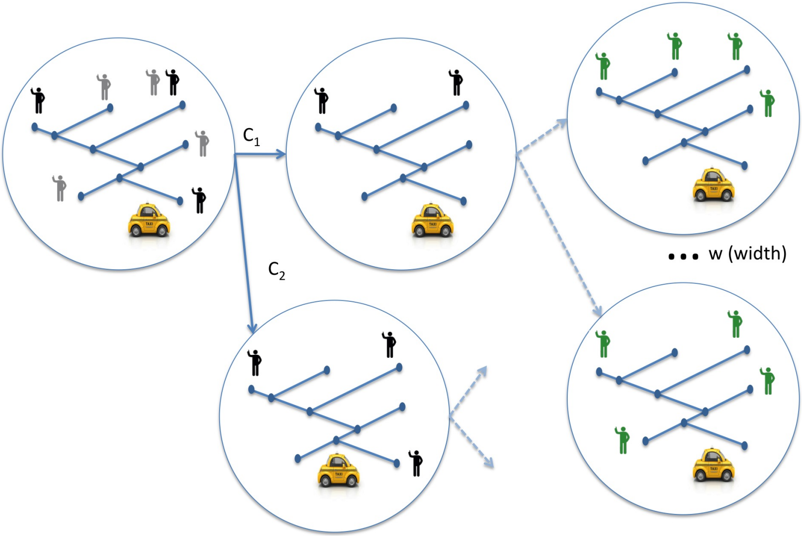
Action Selection algorithms using PSP-HSD

- Reactive (R)
- Hindsight Optimization (HO) [Burns *et al.*, 2012]
- Goal-Distribution-Sensitive (GDS) Planning
    - Step Execution (GDS-SE)
    - Long-term Execution (GDS-LE)

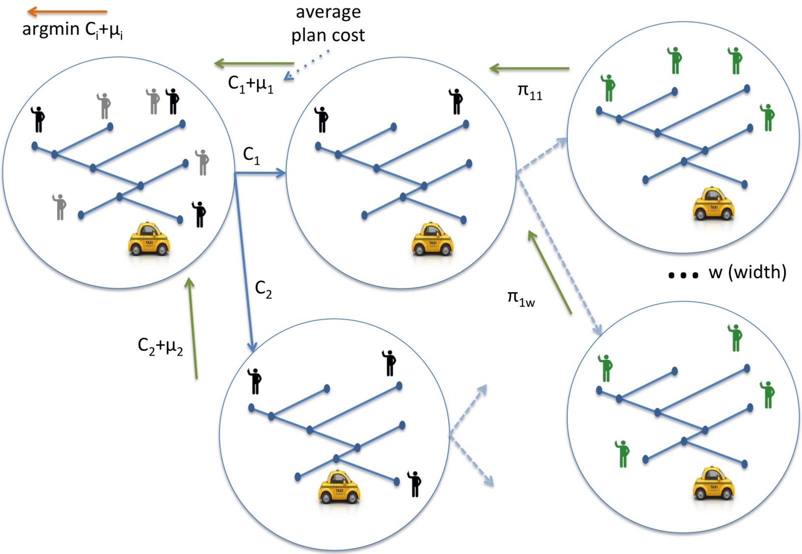
Reactive



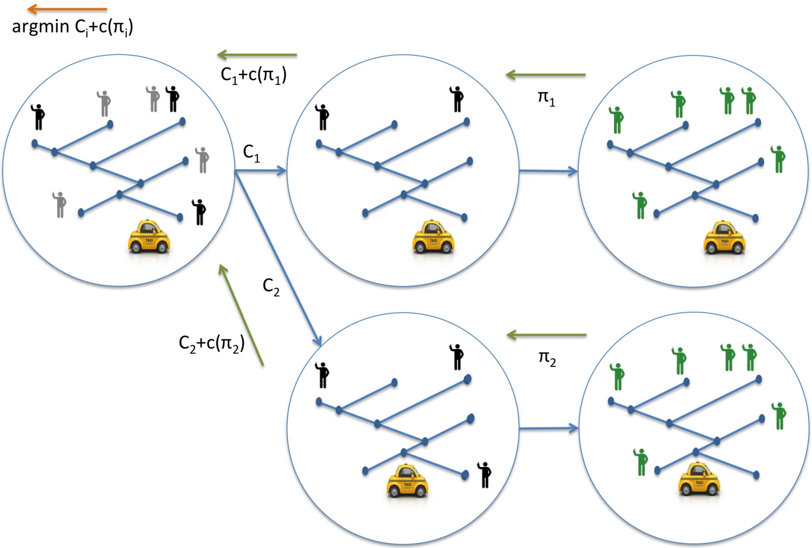
Hindsight Optimization



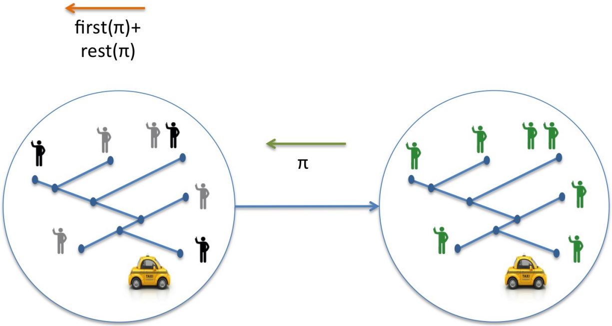
Hindsight Optimization



Short-term Execution



Long-term Execution



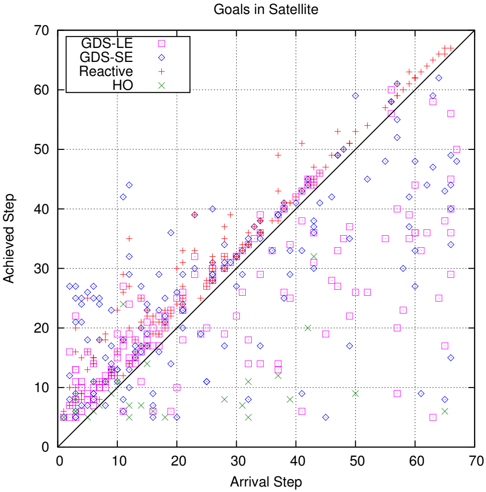
Execution

Technique	Replanning	Future goals
Reactive (GDS-R)	when new goals appear	no
Hindsight Optimization (HO)	at each time step	sampling + determinization
Step Execution (GDS-SE)	at each time step	determinization + state-dependent cost
Long-term Execution (GDS-LE)	when new goals appear	determinization + state-dependent cost

Experimental results. Random Goal Arrival Distribution

Problem (# goals)	**GDS-LE** time	cost	**GDS-SE** time	cost	**HO** time	cost	**Reactive** time	cost
rover-1 (14)	**5.6**	**2.3**	68.8	3.9	22.5	20.4	6.7	3.7
rover-2 (22)	**9.3**	12.5	125.3	34.7	57.2	77.7	11.1	**10.0**
rover-3 (26)	**9.2**	14.6	160.6	41.1	87.7	92.3	12.7	**9.2**
satel-1 (32)	9.7	**4.2**	66.6	**7.3**	27.0	51.0	13.6	7.7
satel-2 (50)	**18.0**	20.1	113.1	41.6	57.7	153.1	22.7	**19.2**
satel-3 (72)	**22.8**	41.9	129.1	127.7	89.9	253.4	30.1	**30.9**
tpp-1 (12)	**5.0**	3.4	52.5	**2.6**	19.4	15.5	**5.0**	4.0
tpp-2 (18)	**7.3**	9.0	87.7	**6.2**	38.6	35.3	7.4	7.2
tpp-3 (24)	**10.1**	18.9	124.5	17.1	80.7	65.2	11.3	**12.3**
uav-1 (24)	**8.1**	5.0	67.0	**4.9**	10.6	35.9	9.7	7.3
uav-2 (40)	**15.6**	16.7	121.3	**13.2**	20.5	125.8	18.2	16.7
uav-3 (60)	**22.2**	59.8	181.5	63.7	68.1	286.4	27.4	**21.9**

Experimental results. Goal arrival vs. achievement



Experimental results. Different goal penalties

Problem	GDS-LE		GDS-SE		Reactive	
	time	cost	time	cost	time	cost
rovers-2	**9.1**	**13.5**	69.2	23.6	9.9	35.3
satel-2	**16.9**	**16.8**	64.7	24.5	20.9	41.2
tpp-2	8.3	44.4	52.0	**34.7**	**8.0**	66.8
uav-2	**15.2**	20.1	65.7	**19.5**	16.1	66.0

Contributions

- Characterization of the PSP-HSD task
  - soft-goals
  - finite horizon
  - state-dependent costs
- Redefinition of two previous action selection schemes
- Definition of two new action selection schemes
- Compilation from PSP-HSD to PDDL

Contributions

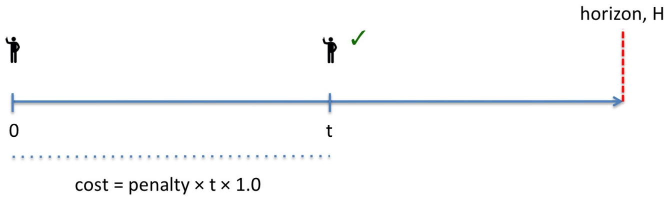
- Characterization of the PSP-HSD task
  - soft-goals
  - finite horizon
  - state-dependent costs
- Redefinition of two previous action selection schemes
- Definition of two new action selection schemes
- Compilation from PSP-HSD to PDDL
- If you ever wonder where do future goals some from,

Contributions

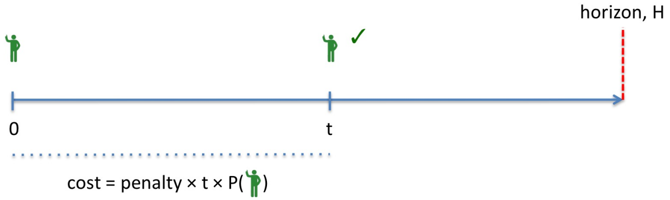
- Characterization of the PSP-HSD task
  - soft-goals
  - finite horizon
  - state-dependent costs
- Redefinition of two previous action selection schemes
- Definition of two new action selection schemes
- Compilation from PSP-HSD to PDDL
- If you ever wonder where do future goals some from, wait until Alberto tells you how to do it...

Thank you

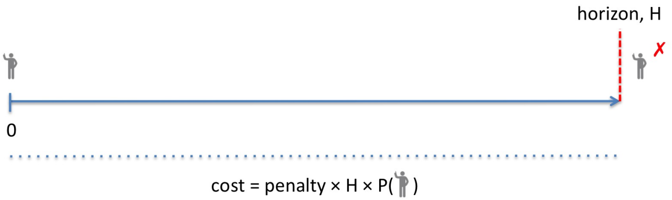
Cost computation



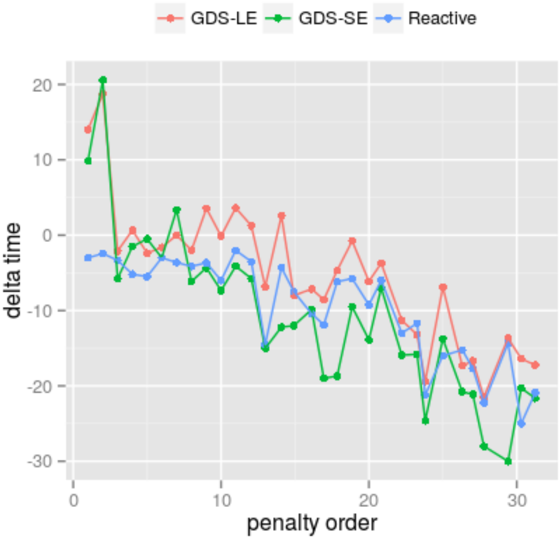
Cost computation



Cost computation



Experimental results. Different goal penalties



Experimental results. Medium-size problems. 300s

Problem (# goals)	**GDS-LE**		**GDS-SE**		**Reactive**	
	time	cost	time	cost	time	cost
rovers-2	**35.0**	**5.3**	595.6	8.9	49.0	7.8
satel-2	**88.0**	**16.0**	562.8	20.8	110.0	18.5
tpp-2	**44.0**	8.6	404.2	**6.8**	45.5	10.3
uav-2	**70.5**	14.3	600.4	**12.0**	84.5	15.5

References

Ethan Burns, J. Benton, Wheeler Ruml, Sungwook Yoon, and Minh B. Do.
Anticipatory on-line planning.
In *Proceedings of ICAPS*, 2012.