Lagrangian Decomposition for Classical Planning (Extended Abstract)

Florian Pommerening[1] Gabriele Röger[1] Malte Helmert[1]
Hadrien Cambazard[2] Louis-Martin Rousseau[3]
Domenico Salvagnin[4]

[1]University of Basel, Switzerland

[2]Univ. Grenoble Alpes, CNRS, Grenoble INP*, G-SCOP, 38000 Grenoble, France
*Institute of Engineering Univ. Grenoble Alpes

[3]Polytechnique Montreal, Canada

[4]University of Padua, Italy

January, 2021

Alternative Talks

IJCAI 2020 long talk (this one)

- high-level ideas in more detail

IJCAI 2020 short talk

- only main idea

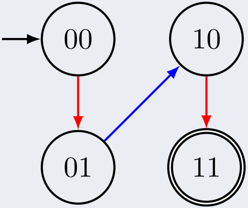
ICAPS 2019

- technical details
- `https://videos.icaps-conference.org`
- `https://ai.dmi.unibas.ch/people/pommeren`

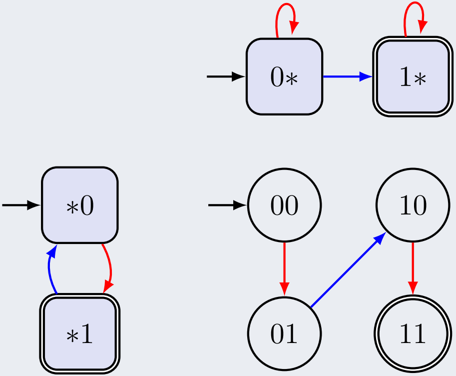
IJCAI Poster presentation

- any level of detail

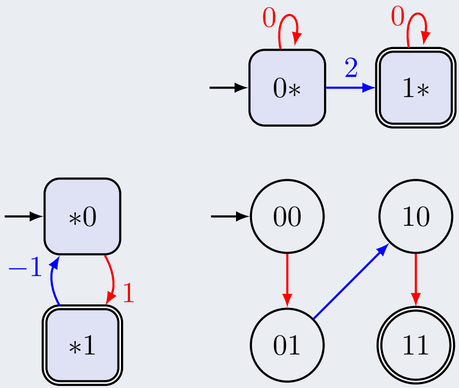
Planning



Abstraction Heuristics



Cost Partitioning



Heuristic value: $2 + 1 = 3$

Lagrangian Relaxation

Original Problem

Minimize objective subject to

$$Ax \geq b$$

other constraints

Lagrangian Relaxation

Original Problem

Minimize objective subject to

$$Ax \geq b$$

other constraints

Lagrangian Subproblem $P(\lambda)$

Minimize objective $+ \lambda(b - Ax)$ subject to

other constraints

Lagrangian Relaxation

Original Problem

Minimize objective subject to

$$Ax \geq b$$

other constraints

Lagrangian Subproblem $P(\lambda)$

Minimize objective $+ \lambda(b - Ax)$ subject to

other constraints

Lagrangian Dual

Maximize $\phi(\lambda)$ subject to

$$\lambda \geq 0$$

Lagrangian Decomposition

- OR technique for LPs with specific form
  - one complicating constraint
  - many independent subproblem constraints
- Lagrangian subproblem decomposes
  - sum of independent subproblems
  - one set of penalty terms per subproblem

Relation to Cost partitioning

Interpretation of Cost Partitioning

original problem ⇔ operator-counting LP (multiple abstractions)

Lagrangian dual ⇔ cost partitioning

Lagrangian multipliers λ_i ⇔ partitioned cost functions

subproblem $P_i(\lambda)$ ⇔ operator-counting LP (single abstraction)

Why is this useful?

Lagrangian dual is continuous and convex

- can use subgradient method
- update multipliers following a subgradient

General Relation to Lagrangian Decomposition

Subgradient g at λ $\Leftrightarrow$ optimal solution of subproblem $P_i(\lambda)$

Cost Partitioning of Abstractions

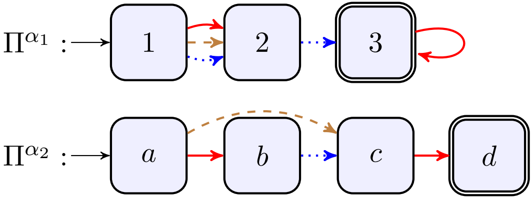
Subgradient g at λ $\Leftrightarrow$ number of times each operator is used in a cheapest plan under cost λ

Subgradient Optimization for Cost Partitioning

Anytime algorithm

- choose any cost partitioning $\mathit{cost}^{(1)}$
- repeat for $t = 1, 2 \ldots$
  - for each abstraction i
    - find optimal solution π^* under $\mathit{cost}_i^{(t)}$
    - set $\mathit{cost}_i^{(t+1)}(o) = \mathit{cost}_i^{(t)}(o) + \eta(t)\mathit{occurrences}(o, \pi^*)$
  - project $\mathit{cost}^{(t+1)}$ to a cost partitioning

Example



Cost in α_1

red solid arrow	brown dashed arrow	blue dotted arrow
0.5	0.5	0.5

Cost in α_2

red solid arrow	brown dashed arrow	blue dotted arrow
0.5	0.5	0.5

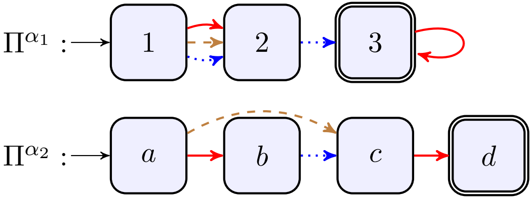
Cheapest Plan in α_1

Cheapest Plan in α_2

Gradient for α_1

Gradient for α_2

Example



Cost in α_1

0.5 0.5 0.5

Cost in α_2

0.5 0.5 0.5

Cheapest Plan in α_1

⟨  ⟩

Cheapest Plan in α_2

⟨  ⟩

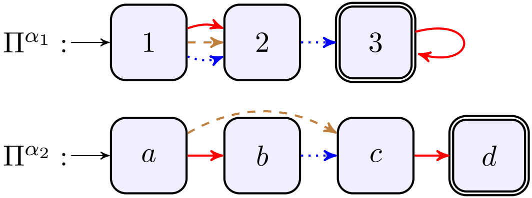
Gradient for α_1

$\langle 1, 0, 1 \rangle$

Gradient for α_2

$\langle 1, 1, 0 \rangle$

Example



Cost in α_1

1.5 0.5 1.5

Cost in α_2

1.5 1.5 0.5

Cheapest Plan in α_1

Cheapest Plan in α_2

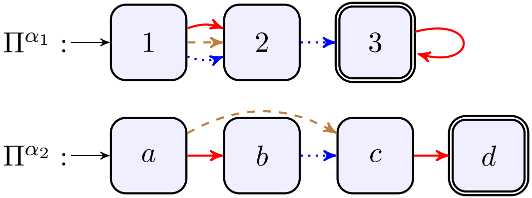
Gradient for α_1

$\langle 1, 0, 1 \rangle$

Gradient for α_2

$\langle 1, 1, 0 \rangle$

Example



Cost in α_1

→	- - →	⋯→
0.5	0	1

Cost in α_2

→	- - →	⋯→
0.5	1	0

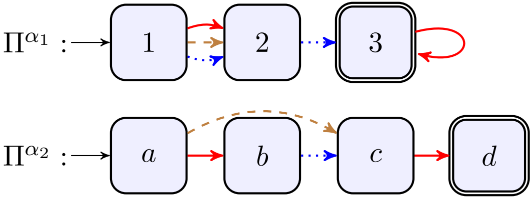
Cheapest Plan in α_1

Cheapest Plan in α_2

Gradient for α_1

Gradient for α_2

Example



Cost in α_1

→	⇢	⋯→
0.5	0	1

Cost in α_2

→	⇢	⋯→
0.5	1	0

Cheapest Plan in α_1

⟨⇢, ⋯→⟩

Cheapest Plan in α_2

⟨→, ⋯→, →⟩

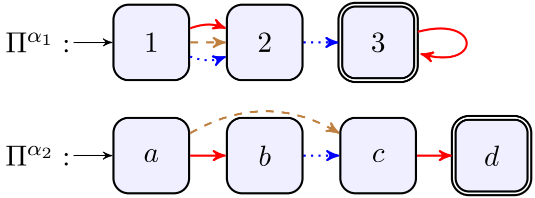
Gradient for α_1

$\langle 0/2, 1/2, 1/2 \rangle$

Gradient for α_2

$\langle 2/2, 0/2, 1/2 \rangle$

Example



Cost in α_1

0.5 0.5 1.5

Cost in α_2

1.5 1 0.5

Cheapest Plan in α_1

Cheapest Plan in α_2

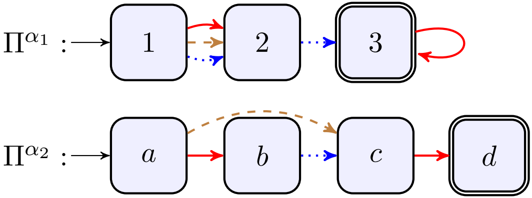
Gradient for α_1

$\langle 0/2, 1/2, 1/2 \rangle$

Gradient for α_2

$\langle 2/2, 0/2, 1/2 \rangle$

Example



Cost in α_1

0	0.25	1

Cost in α_2

1	0.75	0

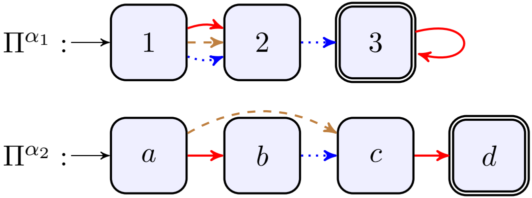
Cheapest Plan in α_1

Cheapest Plan in α_2

Gradient for α_1

Gradient for α_2

Example



Cost in α_1

→	⇢	⋯→
0	0.25	1

Cost in α_2

→	⇢	⋯→
1	0.75	0

Cheapest Plan in α_1

$\langle$ →, ⋯→ $\rangle$

Cheapest Plan in α_2

$\langle$ ⇢, → $\rangle$

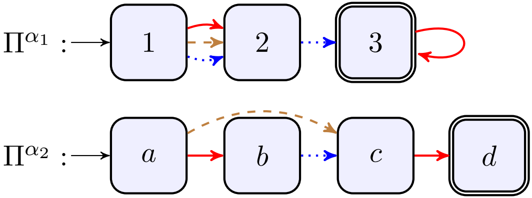
Gradient for α_1

$\langle 1/3, 0/3, 1/3 \rangle$

Gradient for α_2

$\langle 1/3, 1/3, 0/3 \rangle$

Example



Cost in α_1

$0.\bar{3}$ 0.25 $1.\bar{3}$

Cost in α_2

$1.\bar{3}$ $1.08\bar{3}$ 0

Cheapest Plan in α_1

Cheapest Plan in α_2

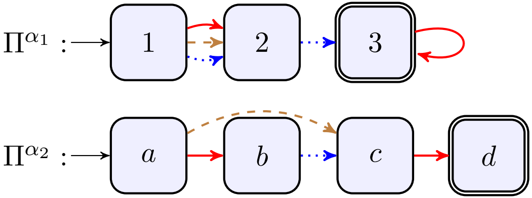
Gradient for α_1

$\langle 1/3, 0/3, 1/3 \rangle$

Gradient for α_2

$\langle 1/3, 1/3, 0/3 \rangle$

Example



Cost in α_1

0	$0.08\overline{3}$	1

Cost in α_2

1	$0.91\overline{6}$	0

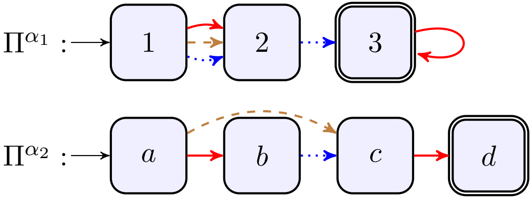
Cheapest Plan in α_1

Cheapest Plan in α_2

Gradient for α_1

Gradient for α_2

Example



Cost in α_1

→	- - →	⋯→
0	$0.08\overline{3}$	1

Cost in α_2

→	- - →	⋯→
1	$0.91\overline{6}$	0

Cheapest Plan in α_1

⟨→, ⋯→⟩

Cheapest Plan in α_2

⟨- - →, →⟩

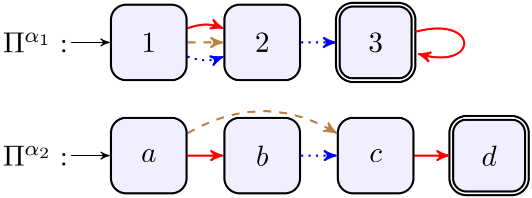
Gradient for α_1

$\langle 1/4, 0/4, 1/4 \rangle$

Gradient for α_2

$\langle 1/4, 1/4, 0/4 \rangle$

Example



Cost in α_1

→	⇢	⋯→
0.25	$0.08\overline{3}$	1.25

Cost in α_2

→	⇢	⋯→
1.25	$1.1\overline{6}$	0

Cheapest Plan in α_1

⟨→, ⋯→⟩

Cheapest Plan in α_2

⟨⇢, →⟩

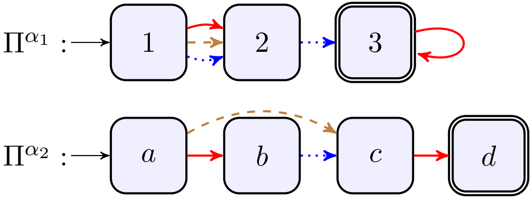
Gradient for α_1

$\langle 1/4, 0/4, 1/4 \rangle$

Gradient for α_2

$\langle 1/4, 1/4, 0/4 \rangle$

Example



Cost in α_1

0 0 1

Cost in α_2

1 1 0

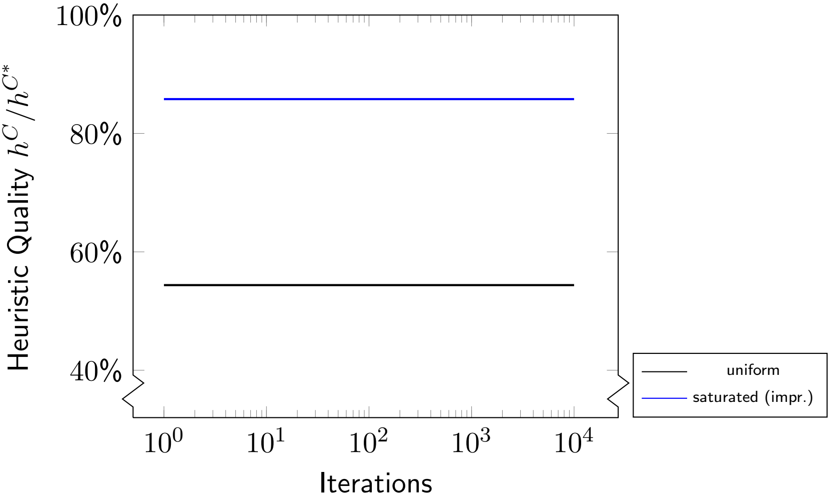
Cheapest Plan in α_1

Cheapest Plan in α_2

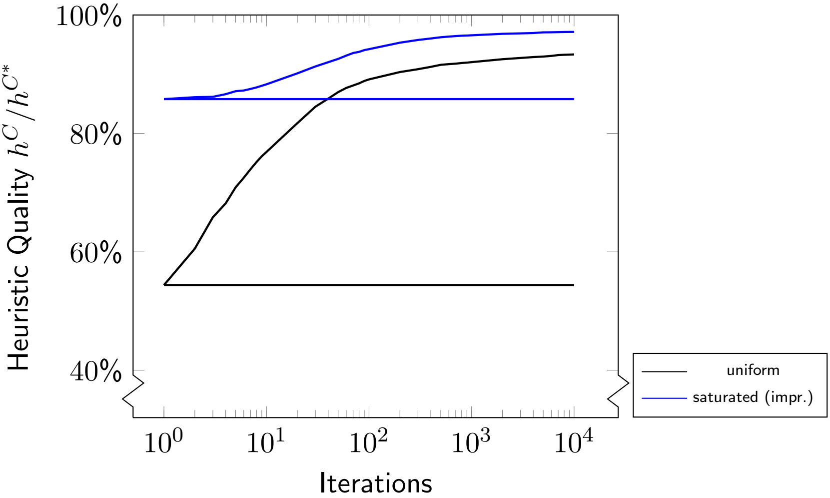
Gradient for α_1

Gradient for α_2

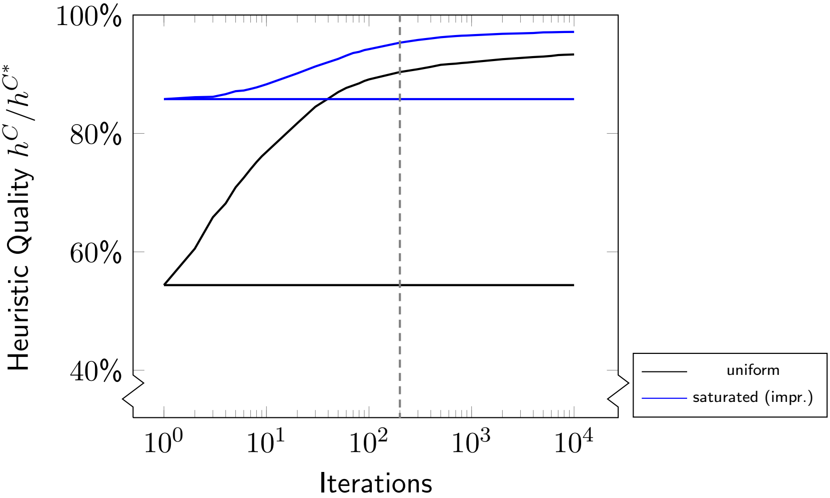
Heuristic Quality



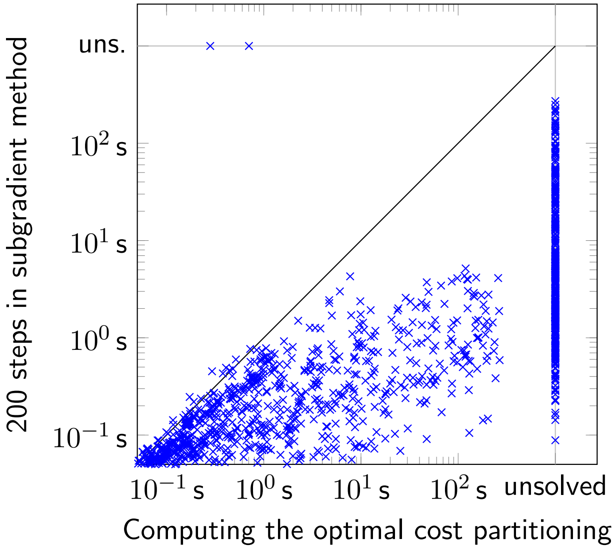
Heuristic Quality



Heuristic Quality



Runtime



uns.
10^2 s
10^1 s
10^0 s
10^{-1} s
200 steps in subgradient method
10^{-1} s
10^0 s
10^1 s
10^2 s
unsolved
Computing the optimal cost partitioning

Conclusion

Main Result

anytime algorithm for computing
optimal cost partitioning without an LP solver

Future Work

explore more options from convex optimization

- stopping condition
- deflection techniques reduce zig-zagging
- step-length updates
- preserved multipliers