Generalized Label Reduction for Merge-and-Shrink Heuristics

Silvan Sievers and Martin Wehrle and Malte Helmert

University of Basel, Switzerland

June 22, 2014

UNI BASEL

Outline

Merge-and-Shrink Heuristic

Computation of merge-and-shrink heuristics:

- Start with the set of atomic transition systems
- Repeatedly apply one of the following:
  - Merge: replace two transition systems by their synchronized product
  - Shrink: replace a transition system by an abstract transition system
- Stop when one transition system is left, use as heuristic

State-of-the-art abstraction heuristic for planning

Outline

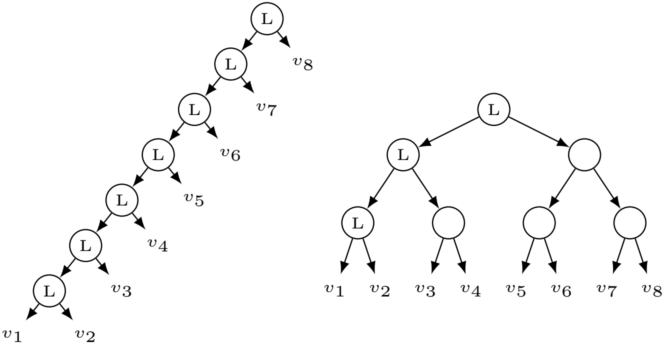
Concept

Label Reduction:

- Identify and eliminate semantically equivalent labels in transition systems
- Always useful:
  - Reduction of memory and time consumption
  - Heuristic quality preserved
  - Fast to compute
- Crucial for efficiently computing merge-and-shrink heuristics

Previous Label Reduction in the Merge-and-Shrink Computation

Example merge trees:



Previous theory:

- Choose one pivot variable
- Label reduction only allowed for transition systems containing pivot variable

Drawbacks

Main drawback of previous label reduction:

- Label reduction limited to one branch of the merge tree

Consequences:

- Usage of linear merge strategies to circumvent drawbacks
- Large part of the space of possible merge strategies not yet explored

Outline

Generalized Label Reduction

Definition

A label reduction for a set of transition systems with label set L is defined as follows:

- For a set of labels $L' \subseteq L$, choose new label $\ell \notin L$ and set $cost(\ell) := min_{\ell' \in L'} cost(\ell')$.
- Replace each label $\ell' \in L'$ by the new label ℓ in all transition systems.

Formally: a label reduction τ is a label mapping, i. e. a function defined on L.

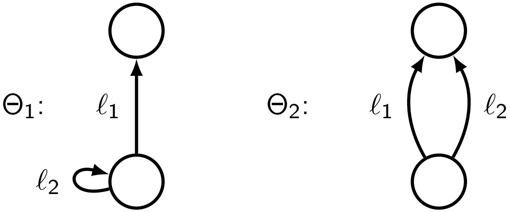
Generalized Label Reduction

Definition

A label reduction for a set of transition systems with label set L is defined as follows:

- For a set of labels $L' \subseteq L$, choose new label $\ell \notin L$ and set $cost(\ell) := min_{\ell' \in L'} cost(\ell')$.
- Replace each label $\ell' \in L'$ by the new label ℓ in all transition systems.

Formally: a label reduction τ is a label mapping, i. e. a function defined on L.



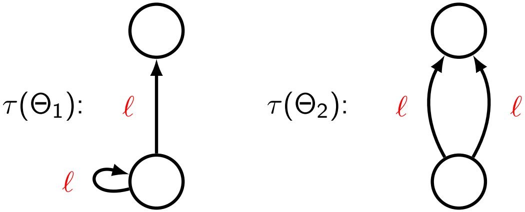
Generalized Label Reduction

Definition

A label reduction for a set of transition systems with label set L is defined as follows:

- For a set of labels $L' \subseteq L$, choose new label $\ell \notin L$ and set $cost(\ell) := min_{\ell' \in L'} cost(\ell')$.
- Replace each label $\ell' \in L'$ by the new label ℓ in all transition systems.

Formally: a label reduction τ is a label mapping, i. e. a function defined on L.



Theorem: Safeness

Theorem

Label reduction is always safe, i. e. leaves the heuristic admissible.
(Formal proof in the paper)

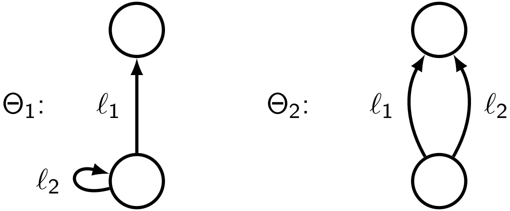
Intuition:

- Transitions are preserved: transitions not lost in synchronized product
- (Goal) states of transition systems not modified
- Transition costs not increased

Combinable Labels

Definition

Let X be a set of transition systems with label set L, let $\ell_1, \ell_2 \in L$ and let $\Theta \in X$.

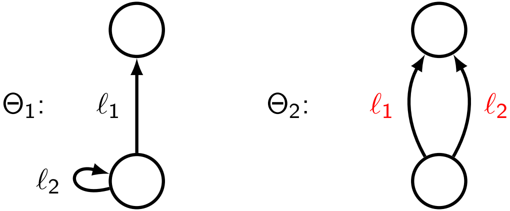


Combinable Labels

Definition

Let X be a set of transition systems with label set L, let $\ell_1, \ell_2 \in L$ and let $\Theta \in X$.

- ℓ_1 and ℓ_2 are locally equivalent in Θ if they label the same set of transitions in Θ.

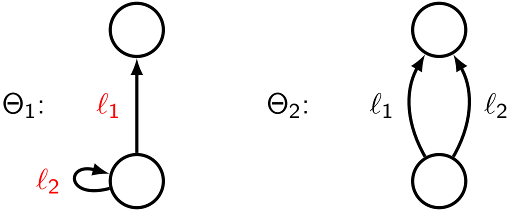


Combinable Labels

Definition

Let X be a set of transition systems with label set L, let $\ell_1, \ell_2 \in L$ and let $\Theta \in X$.

- ℓ_1 and ℓ_2 are locally equivalent in Θ if they label the same set of transitions in Θ.
- ℓ_1 and ℓ_2 are Θ-combinable in X if they are locally equivalent in all $\Theta' \in X \setminus \{\Theta\}$.

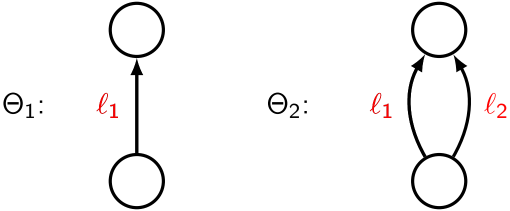


Combinable Labels

Definition

Let X be a set of transition systems with label set L, let $\ell_1, \ell_2 \in L$ and let $\Theta \in X$.

- ℓ_1 and ℓ_2 are locally equivalent in Θ if they label the same set of transitions in Θ.
- ℓ_1 and ℓ_2 are Θ-combinable in X if they are locally equivalent in all $\Theta' \in X \setminus \{\Theta\}$.
- ℓ_1 globally subsumes ℓ_2 if the set of transitions labeled by ℓ_2 is a subset of the set of transitions labeled by ℓ_1 in all transition systems.



Theorem: Exactness

Theorem

Let τ be a label reduction which maps labels ℓ_1 and ℓ_2 onto a new label ℓ.
τ is exact, i. e. leaves the heuristic perfect, iff $cost(\ell_1) = cost(\ell_2)$ and

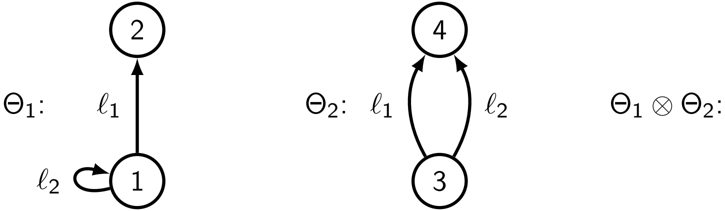
1. *ℓ_1 globally subsumes ℓ_2, or*
2. *ℓ_2 globally subsumes ℓ_1, or*
3. *ℓ_1 and ℓ_2 are Θ-combinable for some $\Theta \in X$.*

Theorem: Exactness

Theorem

Let τ be a label reduction which maps labels ℓ_1 and ℓ_2 onto a new label ℓ.
τ is exact, i. e. leaves the heuristic perfect, iff $cost(\ell_1) = cost(\ell_2)$ and

1. *ℓ_1 globally subsumes ℓ_2, or*
2. *ℓ_2 globally subsumes ℓ_1, or*
3. *ℓ_1 and ℓ_2 are Θ-combinable for some $\Theta \in X$.*

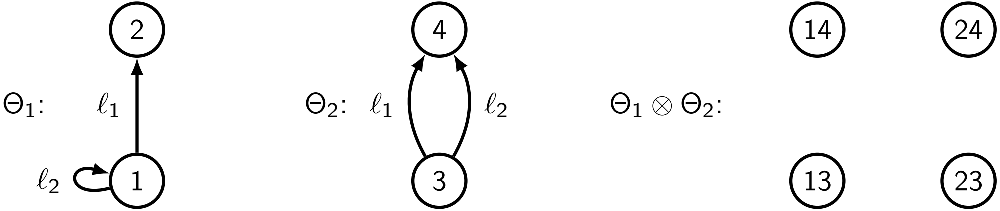


Theorem: Exactness

Theorem

Let τ be a label reduction which maps labels ℓ_1 and ℓ_2 onto a new label ℓ.
τ is exact, i. e. leaves the heuristic perfect, iff $cost(\ell_1) = cost(\ell_2)$ and

1. *ℓ_1 globally subsumes ℓ_2, or*
2. *ℓ_2 globally subsumes ℓ_1, or*
3. *ℓ_1 and ℓ_2 are Θ-combinable for some $\Theta \in X$.*

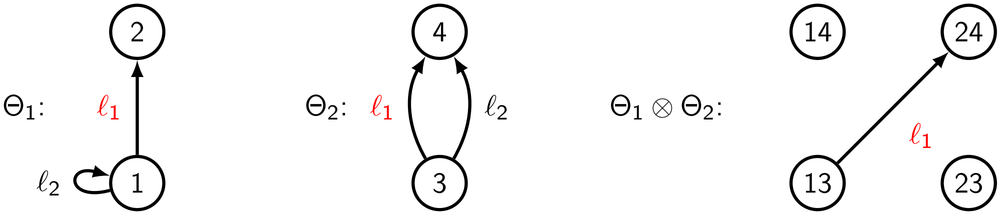


Theorem: Exactness

Theorem

Let τ be a label reduction which maps labels ℓ_1 and ℓ_2 onto a new label ℓ.
τ is exact, i. e. leaves the heuristic perfect, iff $cost(\ell_1) = cost(\ell_2)$ and

1. *ℓ_1 globally subsumes ℓ_2, or*
2. *ℓ_2 globally subsumes ℓ_1, or*
3. *ℓ_1 and ℓ_2 are Θ-combinable for some $\Theta \in X$.*

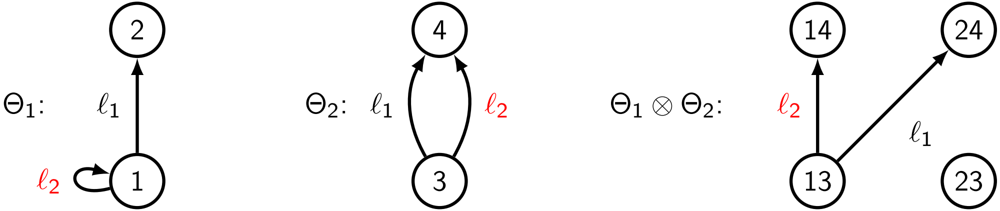


Theorem: Exactness

Theorem

Let τ be a label reduction which maps labels ℓ_1 and ℓ_2 onto a new label ℓ.
τ is exact, i. e. leaves the heuristic perfect, iff $cost(\ell_1) = cost(\ell_2)$ and

1. *ℓ_1 globally subsumes ℓ_2, or*
2. *ℓ_2 globally subsumes ℓ_1, or*
3. *ℓ_1 and ℓ_2 are Θ-combinable for some $\Theta \in X$.*

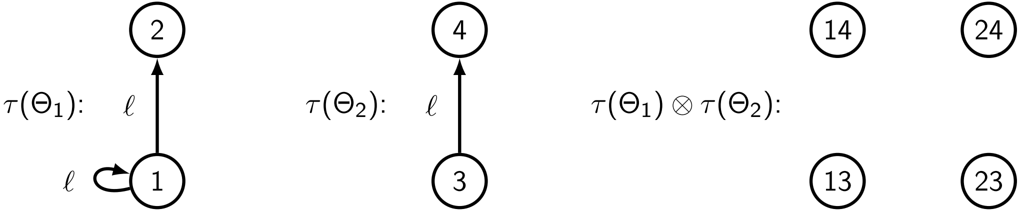


Theorem: Exactness

Theorem

Let τ be a label reduction which maps labels ℓ_1 and ℓ_2 onto a new label ℓ.
τ is exact, i. e. leaves the heuristic perfect, iff $cost(\ell_1) = cost(\ell_2)$ and

1. *ℓ_1 globally subsumes ℓ_2, or*
2. *ℓ_2 globally subsumes ℓ_1, or*
3. *ℓ_1 and ℓ_2 are Θ-combinable for some $\Theta \in X$.*

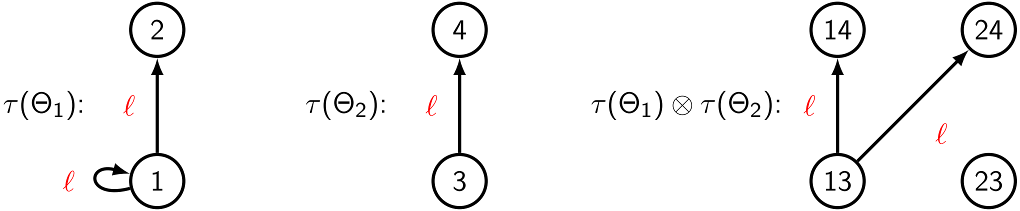


Theorem: Exactness

Theorem

Let τ be a label reduction which maps labels ℓ_1 and ℓ_2 onto a new label ℓ.
τ is exact, i. e. leaves the heuristic perfect, iff $cost(\ell_1) = cost(\ell_2)$ and

1. *ℓ_1 globally subsumes ℓ_2, or*
2. *ℓ_2 globally subsumes ℓ_1, or*
3. *ℓ_1 and ℓ_2 are Θ-combinable for some $\Theta \in X$.*



Outline

Experimental Setup

General:

- Fast Downward planning system

Merge-and-shrink heuristic:

- Linear merge strategy reverse-level (RL)
- Non-linear merge strategy proposed by Dräger et al. (DFP)
- Shrinking based on bisimulation (B)

Coverage Results

Observations:

- Label reduction always useful
- New better than old: larger computational effort compensated by reduced memory/time consumption
- Non-linear merge strategy DFP: best performer

Coverage:

merge/shrink strategy	Label Reduction		
	none	old	new
RL-B-N50k	577	618	634
RL-B-N100k	560	599	639
RL-B-N200k	544	590	630
DFP-B-N50k	565	—	644
DFP-B-N100k	551	—	632
DFP-B-N200k	522	—	625

Outline

Conclusion

Summary:

- Generalized label reduction for merge-and-shrink heuristics:
  - Safe transformation: always allowed on all transition systems
  - Exact transformation: if based on Θ-combinability (among others)
- Prepared the ground for non-linear merge strategies in practice:
  - Implemented non-linear merge strategy DFP
  - Experimental performance gain

The End

Thank you!

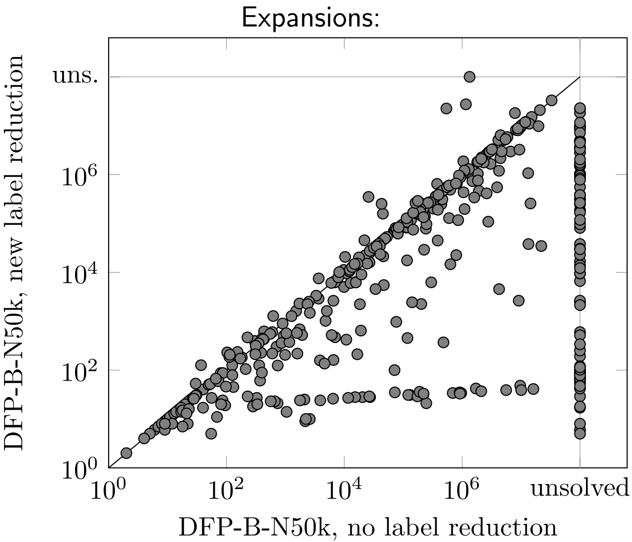
Results: Usefulness of Label Reduction (1)

	RL-B-100K			DFP-B-50K	
	none	old	new	none	new
mprime (35)	8	+6	+15	6	+17
miconic (150)	60	+13	+13	58	+14
gripper (20)	7	+13	+13	7	+11
freecell (80)	6	−2	+13	9	+11
mystery (30)	8	+1	+8	8	+8
zenotravel (20)	9	+3	+3	10	+2
pipesworld-tankage (50)	8	+2	+3	12	+2
nomystery-opt11-strips (20)	17	+1	+1	16	+2
woodworking-opt08-strips (30)	11	−1	+1	11	+2
blocks (35)	25	−3	−3	25	+2
grid (5)	1	+2	+2	1	+1
floortile-opt11-strips (20)	5	+1	+1	4	+1
rovers (40)	7	+1	+1	7	+1
satellite (36)	5	+1	+1	5	+1
scanalyzer-08-strips (30)	12	+1	+1	12	+1
scanalyzer-opt11-strips (20)	9	+1	+1	9	+1
woodworking-opt11-strips (20)	6	−1	+1	6	+1
pipesworld-notankage (50)	14	±0	±0	14	+1
sokoban-opt08-strips (30)	24	±0	+2	25	±0
trucks-strips (30)	6	±0	+2	6	±0
transport-opt11-strips (20)	6	+1	+1	6	±0
driverlog (20)	13	−1	−1	12	±0
Sum (791)	267	+39	+79	269	+79
Remaining domains (605)	293	±0	±0	296	±0
Sum (1396)	560	599	639	565	644

Results: Usefulness of Label Reduction (2)

Remarks:

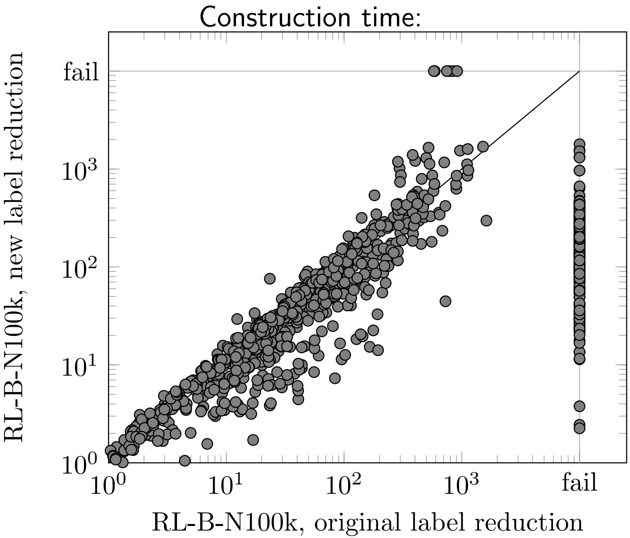
- Label reduction of crucial importance for efficiency
- Bisimulation based shrinking profits from label reduction



Results: Old vs. New Label Reduction Method

Remarks:

- Resulting heuristics similarly informative
- Failures almost always due to memory limit



Previous Label Redcution: Remarks

Weaknesses of previous label reduction:

- Local transformation of one transition system
  (problematic for synchronization behavior)
- Syntax-based comparison of labels
  (requires access to underlying planning operators)
- Independence of shrink strategy
  (no label reduction opportunities from shrinking)

General Label Reduction: Remarks

Notes on the implementation:

- Label reduction through Θ-combinability may enable other Θ'-combinability opportunities
  $\rightarrow$ Label reduction performed as fixpoint computation
- Order of considered transition systems matters
  $\rightarrow$ Randomized order