Merge-and-Shrink: A Compositional Theory of Transformations of Factored Transition Systems

Silvan Sievers   Malte Helmert

University of Basel, Switzerland

ICAPS, June 2022

Merge-and-Shrink

- general class of abstractions
- framework for computing transformations of factored transition systems
- most common use: abstraction heuristics for optimal classical planning
- beyond: proving unsolvability, symbolic search, alternative task representation

Yet Another Paper on Merge-and-Shrink?

previous attempt of a comprehensive theory (journal of the ACM):

- complex dependencies between transformations
- elaborate restrictions on allowed combinations of transformations
- cannot understand properties of merge-and-shrink through properties of its transformations

A New Theoretical Development of Merge-and-Shrink

- compositional theory of transformations of factored transition systems:
  - define desirable properties of transformations such as conservativeness, inducedness, and refinability
  - complete characterization of the conditions under which transformations have these properties
  - composed transformations inherit common properties of component transformations

A New Theoretical Development of Merge-and-Shrink

- compositional theory of transformations of factored transition systems:
  - define desirable properties of transformations such as conservativeness, inducedness, and refinability
  - complete characterization of the conditions under which transformations have these properties
  - composed transformations inherit common properties of component transformations
- first theory on pruning
- first full formal account of factored mappings

Why You Should Read **This** Paper

- almost entirely new theory
- inspired by a line of research by Bäckström & Jonsson

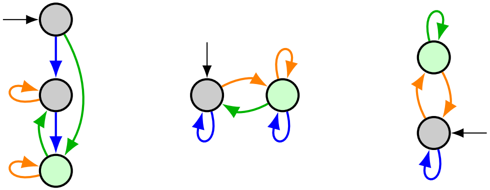
Why You Should Read **This** Paper

- almost entirely new theory
- inspired by a line of research by Bäckström & Jonsson
- learn about a different view of the merge-and-shrink framework
- understand merge-and-shrink transformations and their properties in isolation

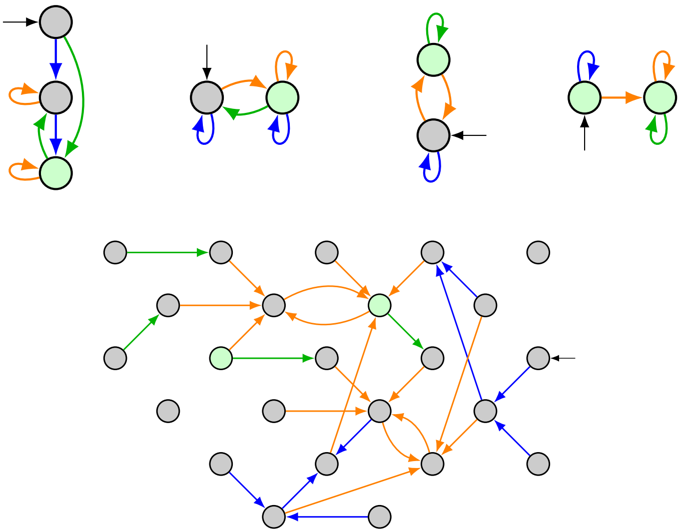
Why You Should Read **This** Paper

- almost entirely new theory
- inspired by a line of research by Bäckström & Jonsson
- learn about a different view of the merge-and-shrink framework
- understand merge-and-shrink transformations and their properties in isolation
- framework applicable beyond computing abstractions
- framework easy to extend with new transformations or new properties

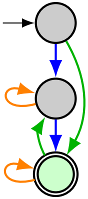
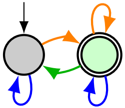
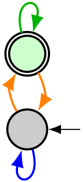
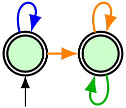
Factored Transition Systems (FTS)



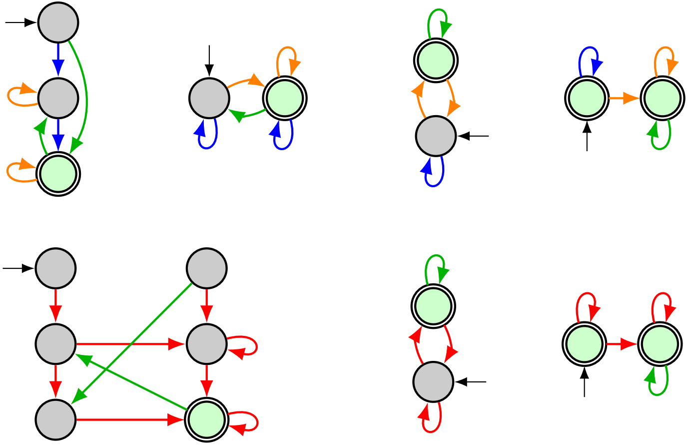
Factored Transition Systems (FTS)



Factored Transformations

Factored Transformations



Properties of Transformations (1)

CONS$_\mathbf{S}$ τ *is* state-conservative *if* $dom(\sigma) = S$*, i.e.,* σ *is a total function.*

CONS$_\mathbf{L}$ τ *is* label-conservative *if* $dom(\lambda) = L$*, i.e.,* λ *is a total function.*

CONS$_\mathbf{C}$ τ *is* cost-conservative *if* $\forall \ell \in L$: $\ell \in dom(\lambda) \rightarrow c'(\lambda(\ell)) \leq c(\ell)$.

CONS$_\mathbf{T}$ τ *is* transition-conservative *if*
$\forall s \xrightarrow{\ell} t \in T$: $s \in dom(\sigma) \wedge t \in dom(\sigma) \wedge \ell \in dom(\lambda) \rightarrow \sigma(s) \xrightarrow{\lambda(\ell)} \sigma(t) \in T'$.

CONS$_\mathbf{I}$ τ *is* initial-state-conservative *if* $\forall s \in S_\mathrm{I}$: $s \in dom(\sigma) \rightarrow \sigma(s) \in S'_\mathrm{I}$.

CONS$_\mathbf{G}$ τ *is* goal-state-conservative *if* $\forall s \in S_\mathrm{G}$: $s \in dom(\sigma) \rightarrow \sigma(s) \in S'_\mathrm{G}$.

IND$_\mathbf{S}$ τ *is* state-induced *if* σ *is surjective, i.e., if* $\forall s' \in S' \; \exists s \in S$: $s \in \sigma^{-1}(s')$.

IND$_\mathbf{L}$ τ *is* label-induced *if* λ *is surjective, i.e., if* $\forall \ell' \in L' \; \exists \ell \in L$: $\ell \in \lambda^{-1}(\ell')$.

IND$_\mathbf{C}$ τ *is* cost-induced *if* $\forall \ell' \in L' \; \exists \ell \in L$: $\ell \in \lambda^{-1}(\ell') \wedge c(\ell) = c'(\ell')$

IND$_\mathbf{T}$ τ *is* transition-induced *if*
$\forall s' \xrightarrow{\ell'} t' \in T' \; \exists s \xrightarrow{\ell} t \in T$: $s \in \sigma^{-1}(s') \wedge t \in \sigma^{-1}(t') \wedge \ell \in \lambda^{-1}(\ell')$.

IND$_\mathbf{I}$ τ *is* initial-state-induced *if* $\forall s' \in S'_\mathrm{I} \; \exists s \in S_\mathrm{I}$: $s \in \sigma^{-1}(s')$.

IND$_\mathbf{G}$ τ *is* goal-state-induced *if* $\forall s' \in S'_\mathrm{G} \; \exists s \in S_\mathrm{G}$: $s \in \sigma^{-1}(s')$.

REF$_\mathbf{C}$ τ *is* cost-refinable *if* $\forall \ell' \in L' \; \forall \ell \in \lambda^{-1}(\ell')$: $c(\ell) = c'(\ell')$.

REF$_\mathbf{T}$ τ *is* transition-refinable *if*
$\forall s' \xrightarrow{\ell'} t' \in T' \; \forall s \in \sigma^{-1}(s') \; \exists s \xrightarrow{\ell} t \in T$: $t \in \sigma^{-1}(t') \wedge \ell \in \lambda^{-1}(\ell')$.

REF$_\mathbf{G}$ τ *is* goal-state-refinable *if* $\forall s' \in S'_\mathrm{G} \; \forall s \in \sigma^{-1}(s')$: $s \in S_\mathrm{G}$.

Properties of Transformations (2)

- conservative transformations: preserve behavior of original FTS

Properties of Transformations (2)

- conservative transformations: preserve behavior of original FTS
- induced transformations: behavior in transformed FTS induced by original FTS

Properties of Transformations (2)

- conservative transformations: preserve behavior of original FTS
- induced transformations: behavior in transformed FTS induced by original FTS
- refinable transformations: every behavior possible in the transformed FTS must be possible in the original one

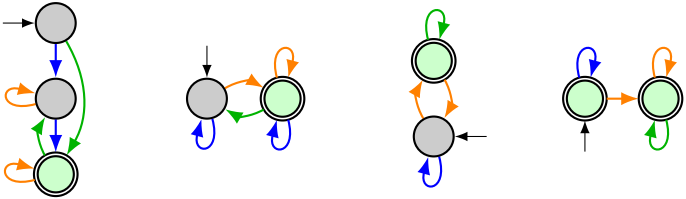
Properties of Transformations (2)

- conservative transformations: preserve behavior of original FTS
- induced transformations: behavior in transformed FTS induced by original FTS
- refinable transformations: every behavior possible in the transformed FTS must be possible in the original one

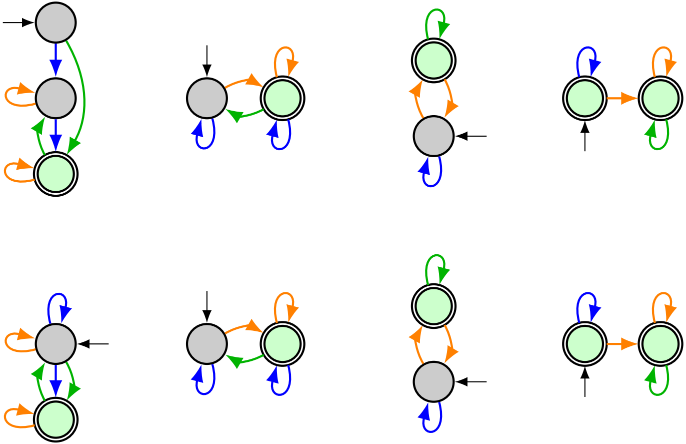
effect on heurisic

- conservative: admissible and consistent heuristics
- conservative + induced: best heuristics among admissible/consistent ones
- exact (= conservative + induced + refinable): perfect heuristics

Shrinking



Shrinking



Shrinking: Properties

- abstraction (conservative + induced)

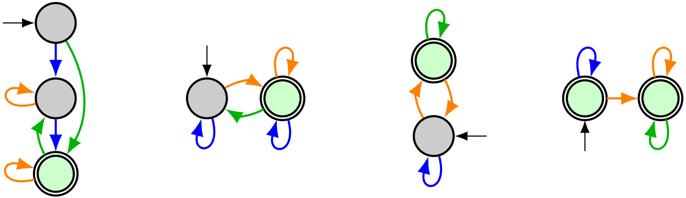
Shrinking: Properties

- abstraction (conservative + induced)

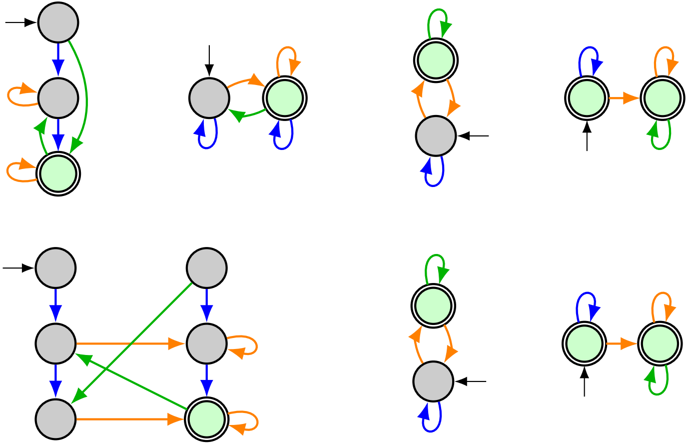
contribution

exact (abstraction + refinable) iff based on bisimulation

Merging



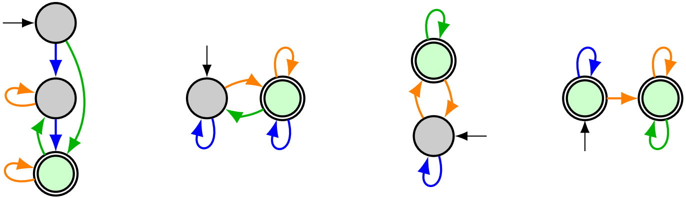
Merging



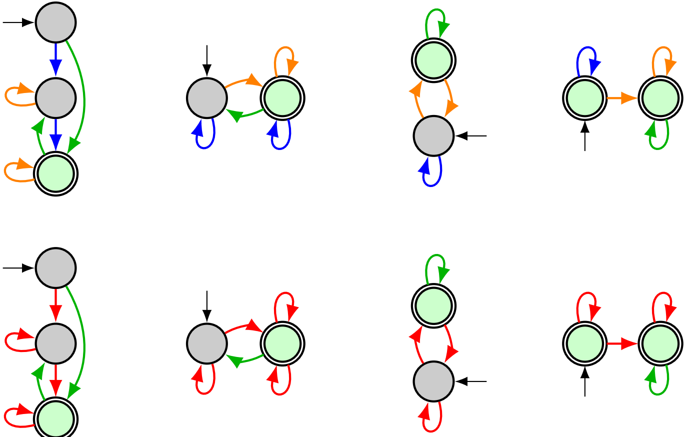
Merging: Properties

- exact

Label Reduction



Label Reduction



Label Reduction: Properties

contribution

- conservative but not induced or refinable in general
- exact iff induced/refinable
- **coNP**-complete to determine if label reduction is induced/refinable

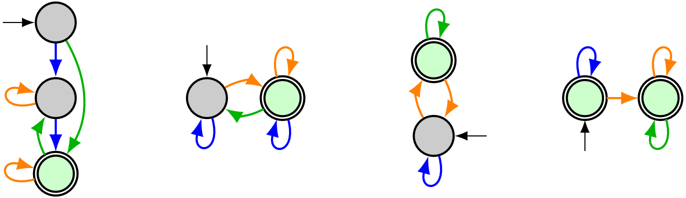
Label Reduction: Properties

contribution

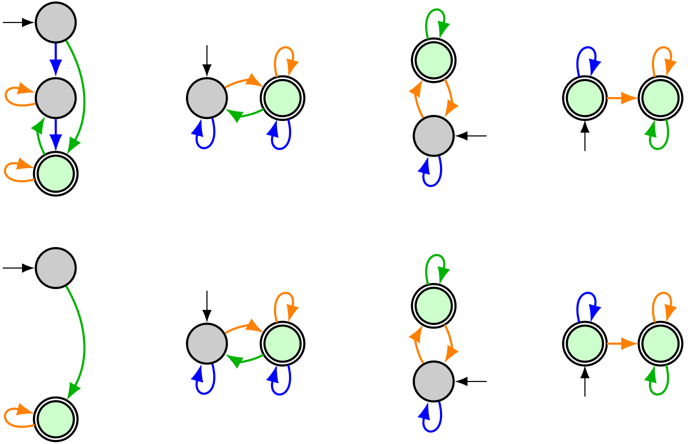
- conservative but not induced or refinable in general
- exact iff induced/refinable
- **coNP**-complete to determine if label reduction is induced/refinable

- atomic label reduction exact iff based on Θ-combinabilty

Pruning



Pruning



Pruning: Properties

contribution

- leads to inadmissible heuristics in general
- exact if keeping exactly the backward-reachable states
- forward-admissible/forward-perfect heuristics if keeping exactly the forward-reachable or alive states

Conclusions

- new theory on merge-and-shrink
- fine-granular properties of transformations
- compositional transformations allow understanding properties of transformations in isolation
- complete characterization of merge-and-shrink transformations