Merge Strategies for Merge-and-Shrink

Master's Thesis

Daniel Federau

13th February 2017

Motivation

- An important factor for the performance of merge-and-shrink is the merge strategy
- There are many merge strategies and improvements described in the literature
- First goal: evaluation of new and existing combinations of merge strategies
- Second goal: implementation of a new combination of MIASM with factored symmetries

Planning Task

- Set of variables with finite domain
- A partial state assigns values to variables
- State is a variable assignment of all variables
- Initial state and set of goal states
- Set of operators with precondition, effect and cost

Transition System

- Planning task induces a transition system (TS)
- Set of states S
- Set of labels L that correspond to operators
- Set of transitions $\langle s, o, s' \rangle \in T$
- Initial state s_0 and set of goal states S_*

Search

- Plan: path in the TS from the initial state to a goal state
- Search: find a plan in the transition system of the planning task
- Optimal Search: plan has the lowest cost among all plans

Heuristic Search

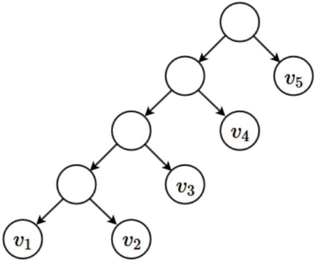
- A heuristic estimates cost from any state to a goal state
- Admissible if heuristic value is never higher than true cost
- Abstraction heuristic: uses cost to goal in abstraction as heuristic value
  - An abstraction maps a TS to a smaller abstract TS
  - An abstract state can correspond to several concrete states

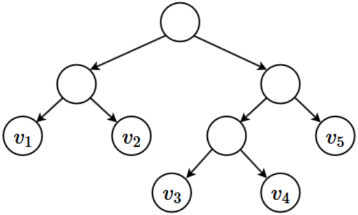
Merge-and-Shrink Algorithm

- manages a set $\mathcal{T}$ of TS
- Start: $\mathcal{T}$ contains an atomic transition systems for every variable
- choose two TS for merging from $\mathcal{T}$
- Shrink step: shrink one or both TS if they are too big
- Merge step: replace the two TS in $\mathcal{T}$ with merge
- Repeat, until only one TS left in $\mathcal{T}$

Merge Strategy

- Decides which two TS will be merged next
- Can be represented by a merge tree
- Linear versus non-linear merge strategy





Linear Merge Strategies

- Causal graph goal level (CGGL)
  - Add variables that are connected to previously added variables in the causal graph
  - Add a variable that has a goal value
- Reverse level (RL) and level (L)
  - Uses the variable level of the causal graph

MIASM

- Goal: merge TS whose product has many unnecessary states
- Unnecessary states:
  - Not reachable from the initial state
  - No path to a goal state
  - Can be pruned
- Subset search of variables
- Partition of variables into subsets
- First merges TS corresponding to variables in each subset
- Then merge remaining TS
- Resulting merge tree is precomputed

Subset search of MIASM

- Best-first search in the space of variable subsets
- Initialisation: add “promising” subsets into priority queue
  - Strongly connected Components (SCC) in the causal graph
  - Mutex groups
- Expand a subset by adding one variable to it
- Subsets are ordered according to formula that uses ratio of necessary states to all states
- Stop if number of states exceeds a defined limit
- Returns a set of subsets that “produce” unnecessary states

DFP and Dynamic-MIASM

- Score for every pair of Transition Systems
- Perform merge with the best score
- DFP:
  - Merges TS that have joint labels that occur near a goal state
- Dynamic-MIASM:
  - Ratio of unnecessary states compared to total amount of states in the merged TS

SCC

- Uses strongly connected components (SCC) of the causal graph
- First merges TS corresponding to variables in SCC
- Then merges all remaining TS
- Uses fall-back strategy for merging

Factored Symmetries

- Factored symmetry:
  - Is a permutation on the set of TS that maps states to states and labels to labels
  - Preserves goal state properties and label costs
- Compute set of TS that are affected by factored symmetry σ
- If available, merge TS that are affected by σ
- If not, merge according to fall-back strategy

Overview

base	CGGL	RL	L	DFP	MIASM	DYN-MIASM
Coverage	710	726	705	745	**756**	744
SYMM-	CGGL	RL	L	DFP	MIASM	DYN-MIASM
Coverage	748	749	741	753	753	**758**
SCC-	CGGL	RL	L	DFP	MIASM	DYN-MIASM
Coverage	743	761	726	**776**	-	762

MIASM and Factored Symmetries

- If factored symmetries are found, merge according to symmetries
- Problem: precomputed merge tree will be ignored
- Goal of MIASM to find unnecessary states not supported

Naive Implementation

- Idea: use factored symmetries in the initialisation of the subset search of MIASM
- Find factored symmetry and merge affected TS
- Replace atomic TS with merged TS in subset search
- Problem: merges without shrinking $\rightarrow$ TS can become too big

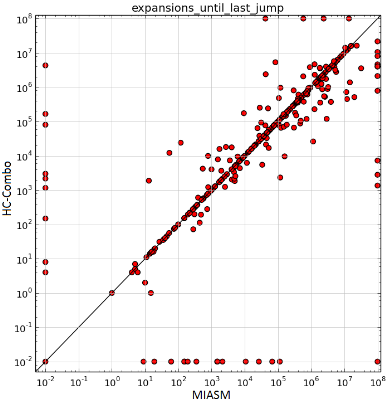
Hill-Climbing Implementation

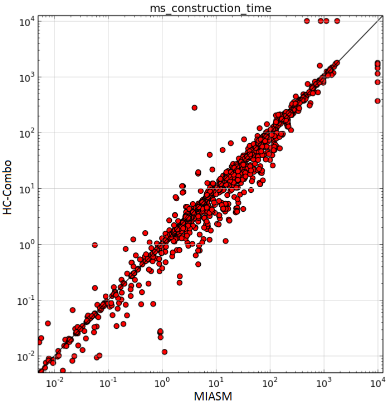
- Idea: only use factored symmetries where all affected TS can be merged efficiently
- Return a set F of TS that are affected by factored symmetries
- Compute all factored symmetries
- Select factored symmetry σ that affects the most TS
- Add set of TS affected by σ to F if it fulfils:
  - Product of the size of the TS affected by σ is smaller than a defined limit
  - All TS affected by σ are disjoint to all TS in F
- Merge all subsets in F for subset search of MIASM

Evaluation - Overview

	MIASM	SYMM-MIASM	Naive-Combo	HC-Combo
Coverage	758	753	667	**768**
Expansions (sum)	417004130	326412007	352085190	**283159812**
M&S Constructions	1444	**1452**	1221	1447
M&S Const. time (avg)	71.02	78.13	71.67	**67.21**
Linear order (%)	70.5	**9.5**	48.4	44.8

Comparison: Original MIASM against Hill Climbing





Conclusion

- Combinations perform better than original strategies
- Only exception is MIASM with symmetries
- Our hill climbing-combination performs better than MIASM and the old combination with factored symmetries
- Factored symmetries do not improve quality but efficiency

Thank you for your attention.

Questions?