Expressing and Exploiting the Common Subgoal Structure of Classical Planning Domains Using Sketches

Dominik Drexler,[1] Jendrik Seipp,[1] Hector Geffner[2,1]

[1]Linköping University, Linköping, Sweden, [2]ICREA &Universitat Pompeu Fabra, Barcelona, Spain
{dominik.drexler, jendrik.seipp}@liu.se, hector.geffner@upf.edu

In a Nutshell

- **Classical planning**
- We consider **tractable planning domains**
- **Policy sketch** defines **subgoal structure**
- **Contribution**: We encode subgoal structure using compact policy sketches to solve whole domains in provably low poly time
- Subproblems are solved with iterated width
- Partial plans are serialized

Iterated Width

- IW(k) is breadth-first search where a newly generated state s is pruned if $novelty(s) > k$
- $novelty(s)$:= smallest size of tuple of atoms made true for the first time
- IW(k) requires $\exp(k)$ time
- When does IW(k) solve a problem?

Width

The **width** $w(P)$ of problem P is the minimum k for which there exists a sequence of atom tuples $t_0, t_1, \ldots, t_m$ each consisting of size at most k atoms, such that

1. t_0 is true in the initial state of P,
2. all opt. plans for t_i can be extended into an opt. plan for t_{i+1} by adding a single action, $i = 1, \ldots, m-1$,
3. if π is an opt. plan for t_m, then π is an opt. plan for P.

- **Theorem:** if $w(P) \leq k$ then IW(k) **solves** P **optimally** in $\exp(k)$ time

The Problem of Unbounded Width

- **Single goal atom** $\Rightarrow$ often small width
- **Conjunctive goals** $\Rightarrow$ often unbounded width
  - **Serialized Iterated Width (SIW)**
  - SIW(k) runs sequence of IW(k) searches
  - Each IW search decreases goal counter $\#g$
  - Subproblems of achieving single goal atom
- **SIW still fails if ...**
  - it traps into an unsolvable state
  - it generates a subproblem of greater width
  - the subproblem has too large width
- Richer decompositions using **policy sketches**

Policy Sketches

- Consider some possibly infinite class of problems Q over some common domain D
- **Policy sketch** (sketch) R defines subgoal structure in every $P \in Q$
- Sketch R is set of **rules** of form $C \mapsto E$ over **features** Φ
- **Sketch width** $w_R(Q)$ is maximum width of all subproblems in all $P \in Q$
- **SIW$_R$** serializes according to subgoals defined by sketch R
- **Theorem:** if $w_R(Q) \leq k$ then SIW$_R$ solves all $P \in Q$ in $\exp(k)$ time

Example Domain: Grid

- Domain description:
  - Robot, key(s), lock(s) distributed in a grid
  - Robot cannot move on a place with closed lock
  - Goal: well place keys; can require opening locks
- SIW generates subproblem of large width
- **Features** $\Phi = \{l, \#g, kl, kg\}$
  - l is number of closed locks
  - $\#g$ is number of wellplaced keys
  - kl whether robot holds key to open lock
  - kg whether robot holds misplaced key
- **Rules** $R_\Phi = \{r_1, r_2, r_3, r_4\}$
  - $r_1 = \{l > 0\} \mapsto \{l\downarrow, \#g?, kl?, kg?\}$
  - $r_2 = \{l = 0, \#g > 0\} \mapsto \{\#g\downarrow, kl?, kg?\}$
  - $r_3 = \{l > 0, \neg kl\} \mapsto \{kl, kg?\}$
  - $r_4 = \{l = 0, \#g > 0, \neg kg\} \mapsto \{kl?, kg\}$
- $w_{R_1}(Q) = 2$ for $R_1 = \{r_1, r_2\}$
- $w_{R_2}(Q) = 1$ for $R_2 = \{r_1, \ldots, r_4\}$

Experiments

	SIW(2)			SIW$_R$(2)			LAMA		Dual-BFWS	
Domain	S	T	MW	S	T	MW	S	T	S	T
Barman (40)	0	–	–	40	0.9	2	40	505.3	40	162.8
Childsnack (20)	0	–	–	20	10.8	1	6	2.6	8	216.9
Driverlog (20)	8	0.5	2	20	0.8	1	20	7.6	20	4.2
Floortile (20)	0	–	–	20	0.2	2	2	9.9	2	176.3
Grid (5)	1	0.1	2	5	0.1	1	5	3.6	5	3.7
Schedule (150)	62	1349.1	2	150	54.7	2	150	15.3	150	151.4
TPP (30)	11	74.7	2	30	0.4	1	30	16.5	29	99.6
# Solved	0/7			7/7			5/7		4/7	