Abstractions for Oversubscription Planning

Vitaly Mirkin
Adviser: Carmel Domshlak
Technion - Israel Institute of Technology
Haifa, Israel

Abstract

In deterministic oversubscription planning, the objective is to achieve as valuable as possible subset of goals within a fixed allowance of the total action cost. Although numerous applications in various fields share this objective, no substantial algorithmic advances have been made beyond the very special settings of net-benefit optimization. Tracing the key sources of progress in classical planning, we identify a severe lack of domain-independent approximations for oversubscription planning, and start with investigating the prospects of abstraction approximations for this problem. In particular, we define the notion of additive abstractions for oversubscription planning, study the complexity of deriving effective abstractions from a rich space of hypotheses, and reveal some substantial, empirically relevant islands of tractability.

Introduction

In the context of deterministic planning, the basic structure of acting in situations with underconstrained or over-constrained resources is respectively captured by classical planning and oversubscription planning. In classical planning, all goals must be achieved at as low a total cost of the actions as possible. In oversubscription planning, an as valuable as possible subset of goals should be achieved within a fixed allowance of the total action cost. While both theory and practice of classical planning have been rapidly advancing, progress in oversubscription planning has been achieved mostly in the special setting of net-benefit planning, where no explicit restriction is put on the plan cost, the action costs and goal utilities are assumed to be comparable, and the objective is thus to maximize the difference between the cumulative value of the achieved goals and the cost invested in achieving them. Research on net-benefit planning resulted in numerous interesting algorithms, but recently it was shown that net-benefit planning is polynomial-time reducible to classical planning (Keyder and Geffner 2009), and thus constitutes an extremely special fragment of oversubscription planning.

A closer look at the recent progress in classical planning reveals that, to a large extent, it stems from advances in domain-independent approximations, or heuristics, of the cost needed to achieve all the goals from a given state. It is thus possible that having a similarly rich pallet of effective heuristic functions for oversubscription planning would advance the state-of-the-art in that setting of automated planning. In principal, the reduction of Keyder and Geffner (2009) from net-benefit to classical planning can be used to reduce oversubscription planning to classical planning with numeric state variables (Fox and Long 2003; Helmert 2002). So far, however, progress in classical planning with numeric state variables has mostly been achieved along variants of monotonic, or delete-free, relaxation heuristics (Hoffmann 2003; Edelkamp 2003), and these heuristics do not preserve information on consumable resources: the “negative” action effects that decrease the values of numeric variables are ignored, possibly up to some special handling of so-called “cyclic resource transfer” (Coles et al. 2008b).

In this work we make first steps towards effective heuristics for oversubscription planning, and in particular, towards admissible abstraction heuristics for this problem. In classical planning, state-space abstractions are among the most prominent foundations for devising admissible heuristics (Edelkamp 2002; Haslum et al. 2007; Helmert, Haslum, and Hoffmann 2007; Katz and Domshlak 2010a). Departing from the most basic question of what state-space abstractions for oversubscription planning actually are (and what they are not), we show that the very notion of abstraction substantially differs in classical and in oversubscription planning. We define additive abstractions and abstraction heuristics for oversubscription planning, and investigate computational complexity of deriving effective abstraction heuristics in the scope of homomorphic abstraction skeletons, paired with cost, value, and budget partitions. Along with revealing some significant islands of tractability, we expose an interesting interplay between knapsack-style problems, convex optimization, and principles borrowed from explicit abstractions for classical planning. We believe that this interplay opens the road to much further research.

Formalism and Background

In line with the SAS+ formalism for deterministic planning (Bäckström and Klein 1991; Bäckström and Nebel 1995), a planning task structure is given by a pair $Π = (V, A)$, where $V$ is a set of $n$ finite-domain state variables, and $A$ is a finite set of actions. Each complete assignment to $V$ is called a state, and $S = \text{dom}(v_1) \times \cdots \times \text{dom}(v_n)$ is the state space of $Π$. Each action $a$ is a pair $⟨\text{pre}(a), \text{eff}(a)⟩$.
of partial assignments to $V$ called preconditions and effects, respectively. Denoting by $\mathcal{V}(p) \subseteq V$ the subset of variables instantiated by a partial assignment $p$, action $a$ is applicable in a state $s$ iff $s[v] = \text{pre}(a)[v]$ for all $v \in \mathcal{V}(\text{pre}(a))$. Applying $a$ changes the value of each $v \in \mathcal{V}(\text{eff}(a))$ to $\text{eff}(a)[v]$. The resulting state is denoted by $s[a]$; by $s[[a_1, \ldots, a_k]]$ we denote the state obtained from sequential application of the (applicable in turn) actions $a_1, \ldots, a_k$ starting at state $s$.

In classical planning, a planning task $\pi = \langle V, A; s_0, G, c, u, b \rangle$ extends its structure with an initial state $s_0 \in S$, a goal specification $G$, typically modeled as a partial assignment to $V$, and a real-valued, nonnegative action cost function $c : A \rightarrow \mathbb{R}^+$. An action sequence $\rho$ is called an $s$-plan if it is applicable in $s$, and $G \subseteq s[\rho]$. An $s$-plan is optimal if the sum of its action costs is minimal among all $s$-plans. The objective in classical planning is to find an $s_0$-plan of as low cost as possible, while optimal classical planning is devoted to searching for optimal $s_0$-plans only.

In contrast, in oversubscription planning, a planning task $\pi = \langle V, A; s_0, G, c, u, b \rangle$ extends its structure with four components: an initial state $s_0 \in S$ and an action cost function $c : A \rightarrow \mathbb{R}^+ \cup \{\infty\}$ as above, plus a succinctly represented and efficiently computable state value function $u : S \rightarrow \mathbb{R}^+$, and a cost budget $b \in \mathbb{R}^+$. An action sequence $\rho$ is called an $s$-plan if it is applicable in $s$, and $\sum_{a \in \rho} c(a) \leq b$. An $s$-plan $\rho$ is optimal if the value of the state reached from $s$ via $\rho$ is maximal among all $s$-plans; in this context, $h^*(s) = \max\{u(s') \mid s' \in S, c(s, s') \leq b\}$ is the value achievable by an optimal $s$-plan in $\Pi$. In the objective in oversubscription planning is to find an $s_0$-plan that brings the system within the cost allowance to as valuable a state as possible, and optimal oversubscription planning is devoted to searching for optimal $s_0$-plans only.

Each planning task $\pi$ induces a state-transition model, often called its transition graph. A transition graph structure (or tg-structure, for short) is a triplet $T = \langle S, L, T \rangle$ where $S$ is the finite set of states, $L$ is the finite set of labels, and $T \subseteq S \times L \times S$ is a set of labeled state transitions. The tg-structure $\pi(T)$ induced by a planning task $\pi = \langle V, A; s_0, G, c, u, b \rangle$ is induced by the structure $(\hat{V}, A)$ of the latter: the states and labels of $\pi(T)$ are states $S = \text{dom}(V)$ and actions $A$ of $\Pi$, respectively, and $(s, a, s[l]) \in T$ if action $a$ is applicable in state $s$.

In oversubscription planning, each tg-structure $T = \langle S, L, T \rangle$ implicitly defines a space of performance measures that can be associated with it. In the context of oversubscription planning, this space constitutes $C \times U \times B$ where $C$ is the set of all nonnegative real-valued functions from labels $L$, $U$ is the set of all nonnegative real-valued functions from states $S$, and $B = \mathbb{R}^+$. A transition graph (or t-graph, for short) $\Phi = \langle T, c, u, b \rangle$ associates a tg-structure $T$ with a specific performance measure $(c, u, b) \in C \times U \times B$, and the t-graph induced by a planning task $\pi = \langle V, A; s_0, G, c, u, b \rangle$ is $\Phi(\pi) = \langle T(\Pi), c, u, b \rangle$.

A few auxiliary notions: For $k \in \mathbb{N}^+$, by $[k]$ we denote the set $\{1, 2, \ldots, k\}$. An $s$-path $(ss^t$-path) in a t-graph $\Phi = \langle T, c, u, b \rangle$ is a path in $T$ from state $s$ (to state $s'$) along the transitions of $T$. For an $ss^t$-path $\pi$, by $u(\pi)$ we refer to the value $u(s')$ of its endpoint. An $s$-path $\pi$ is an $s$-plan for $\Phi$ if $\sum_{(s,l,s') \in \pi} c(l) \leq b$; $\mathcal{P}(s)$ is the set of all $s$-plans for $\Phi$, and $\pi \in \mathcal{P}(s)$ is an optimal $s$-plan for $\Phi$ if $\pi = \arg\max_{\pi' \in \mathcal{P}(s)} u(\pi')$.

For a planning task $\Pi$, searching in its t-graph $\Phi(\Pi)$ corresponds to planning as state-space search. Informed such search procedures employ heuristic functions $h : S \rightarrow \mathbb{R}^+ \cup \{\infty\}$ to estimate the attractiveness of the graph region reachable from a given state $s$. For instance, in classical planning, heuristic $h$ estimates the distance from the given state to the nearest goal state, and it is lower-bounding, or admissible, if $h(s) \leq h^*(s)$ for all states $s$. A useful heuristic function must be both efficiently computable from the planning task, as well as relatively accurate in its estimates. Improving the accuracy of a heuristic function without substantially worsening the time complexity of computing it translates into faster search for plans.

Unlike in oversubscription planning, numerous conceptual and computational ideas in classical planning have been translated to interesting heuristic functions. These ideas include monotonic relaxation (Bonet and Geffner 2001; 2003; Hoffmann and Nebel 2001), critical trees (Haslum and Geffner 2000), logical landmarks for goal reachability (Richter, Helmert, and Westphal 2008; Karpas and Domshlak 2009; Helmert and Domshlak 2009), and abstractions (Edelkamp 2001; Helmert, Haslum, and Hoffmann 2007; Katz and Domshlak 2010a). Also, different heuristics can be combined into their point-wise maximizing and additive ensembles (Edelkamp 2001; Haslum, Bonet, and Geffner 2005; Coles et al. 2008a; Katz and Domshlak 2010b). Unfortunately, while some of these ideas have also been translated to classical planning with numeric state variables, the resulting heuristics do not fit well to the specifics of oversubscription planning.

### Abstractions for Oversubscription Planning

Our focus here is on abstractions for oversubscription planning, from their very definition and properties, to the prospects of deriving abstraction heuristics.

The term “abstraction” is usually associated with simplifying the original system, factoring out details less crucial in the given context. Which details can be factored out and which had better be preserved depends on the context. In classical planning, the abstract t-graphs are required not to increase the distances between the (abstracted) states (Edelkamp 2001; Helmert, Haslum, and Hoffmann 2007). Such “distance conservation” is in particular guaranteed by homomorphic abstractions, obtained by systematically contracting sets of states into single abstract states. In turn, an additive abstraction in classical planning is a set of abstractions, inter-constrained to jointly not overestimate the state-to-state costs of the original task. In contrast, a set of abstractions in oversubscription planning is constrained to jointly not underestimate the value that can be obtained from a concrete state of the original task within a given cost budget. Hence, the notions of additive abstraction for classical and oversubscription planning very different, with the latter providing us, for good and for bad, with many more degrees of freedom.
Definition 1 (Additive Abstraction) Let $\Phi = \langle T, c, u, b \rangle$ be a t-graph with $T = \langle S, L, T_r \rangle$, and $s_0$ be a state in $S$.

- An abstraction skeleton $\mathcal{AS}$ of $T$ is a set of pairs $\mathcal{AS} = \{\mathcal{T}_i, \alpha_i\}_{i \in [k]}$ where $\mathcal{T}_i = \langle S_i, L_i, T_r \rangle$ is a tg-structure and $\alpha_i : S \to S_i$ is a state mapping from $T$ to $T_i$.
- A set of t-graphs $\mathcal{A} = \{\Phi_i\}_{i \in [k]}$ is an abstraction for $s_0$ in $\Phi$ with respect to $\mathcal{AS}$, denoted by $\mathcal{A} \in \mathcal{AS}$, if
  - $\Phi_1 = \langle T_1, c_1, u_1, b_1 \rangle$ and
  - if $\pi$ is an optimal $s_0$-plan for $\Phi$, and for $i \in [k]$, $\pi_i$ is an optimal $\alpha_i(s_0)$-plan for $\Phi_i$, then $\hat{u}(\pi) \leq \sum_{i \in [k]} u_i(\pi_i)$.

To illustrate this definition, consider a tg-structure $T = \langle \{s_1\}_{i \in [5]}, \{l_1\}_{i \in [5]}, T_r \rangle$ depicted in Figure 1a, and an abstraction skeleton $\mathcal{AS} = \{\langle T_1, \alpha_1 \rangle, \langle T_2, \alpha_2 \rangle \}$ of $T$, with t-graphs $T_1, T_2$ as in Figure 1b and state mappings

$$\alpha_1(s_1) = \begin{cases} s_1, & i \in \{2, 4\} \\ s_2, & i = 3 \\ s_3, & \text{otherwise} \end{cases} \quad \alpha_2(s_1) = \begin{cases} s_2, & i \in \{2, 4\} \\ s_3, & \text{otherwise} \end{cases}$$

Let t-graphs $\Phi = \langle T, c, u, b \rangle, \Phi_1 = \langle T_1, c_1, u_1, b_1 \rangle, \Phi_2 = \langle T_2, c_2, u_2, b_2 \rangle$ be defined via label cost functions $c_1, c_2$ that associate all labels with a cost of 1, budgets $b_1 = b_2 = 2$, and state value functions $u_1, u_2$ that evaluate to zero on all states except for the states $s_1, s_2, s_3, s_4$ on which they respectively evaluate to one. Considering the state $s_1$ in $T$, the optimal $s_1$-plan for $\Phi$ is $\pi = \langle (s_1, l_1, s_2), (s_3, l_4, s_5) \rangle$ with $\hat{u}(\pi) = 1$. The optimal $\alpha_1(s_1)$-plan for $\Phi_1$ is $\pi_1 = \langle (s_1, l_1, s_2), (s_1, l_1, s_2) \rangle$ with $\hat{u}_1(\pi_1) = 1$, and the optimal $\alpha_2(s_1)$-plan for $\Phi_2$ is $\pi_2 = \langle (s_1, l_1, s_2), (s_1, l_1, s_2) \rangle$, with $\hat{u}_2(\pi_2) = 1$. Since $\hat{u}(\pi) \leq \hat{u}_1(\pi_1) + \hat{u}_2(\pi_2)$, $\mathcal{A} = \{\Phi_1, \Phi_2\}$ is an additive abstraction for $s_1$ in $\Phi$ with respect to $\mathcal{AS}$.

Definition 2 (Abstraction Heuristics) Let $\Pi$ be a planning task with state space $S$, and $\mathcal{AS} = \{T_i, \alpha_i\}_{i \in [k]}$ be an abstraction skeleton of $\mathcal{T}(\Pi)$.

- For each state $s \in S$, and each additive abstraction $\mathcal{A} = \{\Phi_i\}_{i \in [k]} \in \mathcal{AS}$, $h_A(s) = \sum_{i \in [k]} \hat{u}_i(\pi_i)$ where $\pi_i$ is an optimal $\alpha_i(s)$-plan for $\Phi_i$.
- A function $h : S \to \mathbb{R}^{0+}$ is an AS-heuristic for $\Pi$ if, for each state $s \in S$, $h(s) = h_A(s)$ for some additive abstraction $\mathcal{A} \in \mathcal{AS}$.

Theorem 1 For any planning task $\Pi$, any abstraction skeleton $\mathcal{AS}$ of $\mathcal{T}(\Pi)$, any state $s$ of $\Pi$, and any $\mathcal{A} \in \mathcal{AS}$, $h_A(s) \geq h^*(s)$.

Sub-claim (1) in Theorem 1 is immediate from Definition 2, and it in particular implies that, for any AS-heuristic $h$ for $\Pi$, $h(s) \geq h^*(s)$. The proof of (2) in Theorem 1 is also straightforward: Let $\mathcal{A} = \{\Phi_i = \langle T_i, c_i, u_i, b_i \rangle\}_{i \in [k]}$ be an additive abstraction for $s$ in $\Pi$. For $i \in [k], \alpha_i(S_i) = \{s' \in S_i | c_i(\alpha_i(s)), s' \leq b_i \}$. Since $\mathcal{A}$ is given explicitly, computing shortest paths from $\alpha_i(s)$ to all states in $T_i$, and thus computing $S_i'$, can be done in time polynomial in $|A|$. Figure 1: Illustration for our running example.
show below, even some constrained estimate optimizations of this kind can be challenging. The lattice in Figure 2 depicts the whole range of options for such constrained optimization; at the extreme settings, $H_{(\cdot,-,\cdot)}(s)$ is simply a renaming of $A(s)$, and $h_{(c,u,b)}(s)$ corresponds to a single abstraction $(c,u,b) \in A(s)$.

In a recent work (Domshlak and Mirkus 2013) we present some theoretical and practical initial results.

References


Helmert, M. 2002. Decidability and undecidability results for planning with numerical state variables. In *Proceedings of the Sixth International Conference on Artificial Intelligence Planning and Scheduling*.


