Efficient Evaluation of Large Abstractions for Decoupled Search: Merge-and-Shrink and Symbolic Pattern Databases – Supplementary Material

Daniel Gnad[1], Silvan Sievers[2], Álvaro Torralba[3]

[1] Linköping University, Sweden
[2] University of Basel, Switzerland
[3] Aalborg University, Denmark
daniel.gnad@liu.se, silvan.sievers@unibas.ch, alto@cs.aau.dk

The presented supplementary material contains full proofs for Theorem 1, Corollary 2, and Proposition 5 of the main paper, as well as additional experimental results that evaluate the size of the explored search space.

Full Proofs

Theorem 1. *Let A be an ADD that represents a PDB heuristic h, $s^{\mathcal{F}}$ a decoupled state, and $B \in \mathbb{R}^+ \cup \{\infty\}$ a bound. It is **NP**-complete to decide if $h_{\mathcal{F}}(s^{\mathcal{F}}) < B$.*

Proof. For membership, one can guess a state $s \in [s^{\mathcal{F}}]$ and check if $h(s) < B$ in polynomial time.

For hardness, consider the following planning task encoding 3-SAT. For a 3-CNF formula with propositions X and clauses $\{C_1, \dots, C_m\}$, we construct Π as follows:

- $\mathcal{V} = \{v_g\} \cup \{v_x \mid x \in X\} \cup \{v_x^{L_i}, v_y^{L_i}, v_z^{L_i}, v^{L_i} \mid x, y, z \text{ in } C_i\}$ with domains:
  - $\mathcal{D}(v) = \{u, 0, 1\}$ (unassigned, false, true) for $v \in \{v_w^{L_i} \mid w \text{ in } C_i, 1 \leq i \leq m\} =: V_1$
  - $\mathcal{D}(v) = \{i, g\}$ (initial, goal) for $v \in \{v_g\} \cup \{v_x \mid x \in X\} \cup \{v^{L_i} \mid 1 \leq i \leq m\} =: V_2$
- $I = \{v = u \mid v \in V_1\} \cup \{v = i \mid v \in V_2\}$
- $G = \{v = g \mid v \in V_2\}$

The factoring $\mathcal{F}$ is such that for every clause C_i, there is a leaf factor $L_i = \{v_x^{L_i}, v_y^{L_i}, v_z^{L_i}, v^{L_i}\}$ that contains a variable $v_w^{L_i}$ for every proposition w in C_i and v^{L_i} for book-keeping. The center factor is $C = \{v_g\} \cup \{v_x \mid x \in X\}$.

For every leaf L_i and every assignment that satisfies C_i (there are seven such assignments) there exists an action $a^{L_{ij}}$ ($1 \leq j \leq 7$) that sets v^{L_i} from i to g and sets the propositional variables $v_w^{L_i}$ for all w in C_i from u to the corresponding value. Moreover, there exist $2|X| + 1$ global actions: there are two actions a_x^0 and a_x^1 per $x \in X$ with preconditions $\mathsf{pre}(a_x^0) = \{v_x = i\} \cup \{v_x^{L_i} = 0 \mid \forall L_i : x \text{ in } C_i\}$ and $\mathsf{pre}(a_x^1) = \{v_x = i\} \cup \{v_x^{L_i} = 1 \mid \forall L_i : x \text{ in } C_i\}$ and effect $\mathsf{eff}(a_x^0) = \mathsf{eff}(a_x^1) = \{v_x = g\}$. Finally, there is an action a_g with precondition $\mathsf{pre}(a_g) = \{v^{L_i} = g \mid 1 \leq i \leq m\} \cup \{v_g = i\}$ and effect $\mathsf{eff}(a_g) = \{v_g = g\}$. All actions have cost 0.

In decoupled search, the initial state $I^{\mathcal{F}}$ has a price of 0 for the initial leaf states $\{v^{L_i} = i, v_x^{L_i} = u, v_y^{L_i} = u, v_z^{L_i} = u\}$ and all leaf states that satisfy $v^{L_i} = g$ and that set the propositional variables $v_x^{L_i}, y^{L_i}, z^{L_i}$ to truth values which correspond to a satisfying assignment for x, y, z in clause C_i (these leaf states can be reached via the leaf actions a^{L_i}).

Let $s^{\mathcal{F}}$ be the decoupled state that results from applying a_g to $I^{\mathcal{F}}$. Our proof works by showing that we can compactly represent a PDB heuristic as an ADD that detects if $s^{\mathcal{F}}$ is a dead end, which is true iff the 3-CNF formula is unsatisfiable.

Let h be the PDB heuristic that uses the pattern $P = \{v_x \mid x \in X\} \cup \{v_w^{L_i} \mid w \text{ in } C_i, 1 \leq i \leq m\}$. Observe that an assignment $v_x = g$ implies that all $v_x^{L_i}$ have the same value, because otherwise $a_x^{0/1}$, which sets v_x to g, is not applicable. Therefore, every state s where, for any two clauses C_i, C_j that share a proposition $x \in X$, the values assigned to $v_x^{L_i}$ and $v_x^{L_j}$ *differ* is a dead end in the abstract state space, so $h(s) = \infty$. All other states have a heuristic value of 0.

We next show that the 3-CNF formula is satisfiable if and only if $h_{\mathcal{F}}(s^{\mathcal{F}}) = 0$, and otherwise $h_{\mathcal{F}}(s^{\mathcal{F}}) = \infty$. Observe first that, in $s^{\mathcal{F}}$, all leaf states with $v^{L_i} = i$ are not reached due to the precondition $v^{L_i} = g$ of a_g. That means that the compliant leaf paths of all leaves must involve some leaf action setting v^{L_i} to g, thereby also setting all propositional variables $v_w^{L_i}$ for w in C_i to a truth value. Observe further that every satisfying assignment for all clauses C_i is reached in $s^{\mathcal{F}}$. Finally, $s^{\mathcal{F}}$ is solvable only if for every $x \in X$ there exists a reached leaf state in every L_i that contains x such that all $v_x^{L_i}$ have the same value. This is exactly captured by h and is the case iff the formula is satisfiable.

Therefore, all that remains to show is that we can represent h with an ADD of polynomial size. Indeed, we construct an ADD A using a variable order that keeps next to each other all variables encoding the proposition $x \in X$ in different clauses, and concatenate these clusters using an arbitrary ordering of the propositions:

$$\langle v_x, v_x^{L_1}, \dots, v_x^{L_k}, v_y, v_y^{L_o}, \dots, v_y^{L_p}, v_z, v_z^{L_q}, \dots, v_z^{L_r}, \dots \rangle$$

Figure 1 illustrates a (simplified) part of the ADD, with two gray blocks that correspond to the clusters of variables just described, for two exemplary propositions $x, y \in X$. The dashed edges from a node v_x, for $x \in X$, represent the value g (implying that all $v_x^{L_i}$ have the same value) and the solid edges represent the value i. The ADD in the figure represents h: if $v_x = i$ in the given state, we need to check that all values of $v_x^{L_j}$ for $1 \leq j \leq k$ are equal to ensure that we are in a non-dead-end state; if $v_x = g$, the given state cannot be a dead end due to x and we can continue with y. Hence, from all nodes $v_x^{L_i}$ in the cluster of proposition $x \in X$, edges either go to a node in the cluster, where the left branch (dashed edges) indicates that $v_x^{L_i} = 0$ and the right branch (solid edges) that $v_x^{L_i} = 1$, or go to either ∞ or v_y, where y is the successor of x in the ordering of X.

We remark that, as the variables $v_x^{L_i}$ are non-binary, the illustration of A in Figure 1 is imprecise, assuming that $v_x^{L_i} \neq u$ always holds. This is a simplification without loss of generality for an easier visualization: the proper encoding of the PDB heuristic h described above uses twice the number of nodes, duplicating each $v_x^{L_i}$ node to handle the case $v_x^{L_i} = u$.

We next show that the size of A can be bounded by $O(|X||C|)$. In Figure 1, every cluster of variables $v_x, v_x^{L_1}, \dots, v_x^{L_k}$ for each $x \in X$ is expressed as an ADD that uses $2k \leq 2|C|$ internal nodes, as at every of the k layers, we only need two nodes to distinguish the case where all previous variables were 0 or 1. Note that no node needs to represent the case where two variables differ because in that case, we go directly to the terminal node ∞. The clusters are independent, and therefore the corresponding ADDs are just stacked on top of each other (Edelkamp and Kissmann 2008), like for x and y in the illustration.

So all involved components, the planning task, factoring, and ADD are polynomially bounded by the size of the 3-CNF formula. The claim follows since $s^{\mathcal{F}}$ is detected as a dead-end by $h_{\mathcal{F}}$ iff the formula is satisfiable. □

Corollary 2. *Let σ be an FM that represents a merge-and-shrink heuristic h and $s^{\mathcal{F}}$ a decoupled state. It is **NP**-complete to decide if $h_{\mathcal{F}}(s^{\mathcal{F}}) < B$.*

Proof. Membership follows trivially as before. For hardness, we remark that every ADD can be transformed into an FM in polynomial time (Edelkamp, Kissmann, and Torralba 2012; Torralba 2015). With that, we can take the ADD A constructed in the proof of Theorem 1, transform it into an FM σ, and evaluate the same decoupled state on it. Again, all operations are polynomial in the size of the input 3-CNF formula and $s^{\mathcal{F}}$ is detected as dead-end iff the formula is satisfiable. □

Proposition 5. *Let $\mathcal{F}$ be a factoring and σ an FM. If σ is strongly compliant with $\mathcal{F}$, then σ is compliant with $\mathcal{F}$.*

Proof. We show that σ and all its descendants have a nestedness of at most 1, which implies compliance with $\mathcal{F}$. First, we consider FMs that exactly cover a leaf factor and their descendants, then we consider the remaining FMs.

Let $\Sigma^{\mathcal{L}}$ be the set of descendants σ^L of σ that exactly cover a leaf $L \in \mathcal{L}$ and do not cover any other leaf, i. e., $cov(\sigma^L) = ec(\sigma^L) = \{L\}$. Observe that for all $\sigma^L \in \Sigma^{\mathcal{L}}$ (and all their descendants), we have that $|cov(\sigma^L)| \leq 1$. This implies that all these FMs have nestedness of at most 1, implying compliance with $\mathcal{F}$.

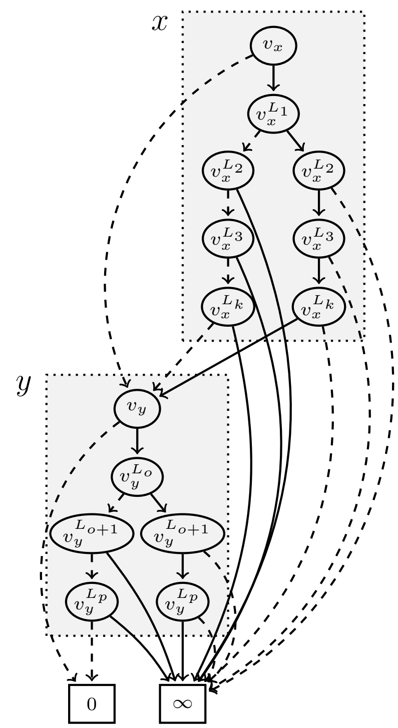

Figure 1: Illustration of a part of the ADD described in the proof of Theorem 1.

It remains to show that FMs that are neither in $\Sigma^{\mathcal{L}}$, nor one of their descendants, are compliant, too. Let $\Sigma^{>1L}$ denote the subset of these FMs σ' where $|cov(\sigma')| > 1$ (again, compliance is implied if $|cov(\sigma')| \leq 1$). Let $\sigma' \in \Sigma^{>1L}$. As σ is strongly compliant, every leaf $L \in cov(\sigma)$ is exactly covered in one of its descendants, excluding σ itself. So every such leaf is covered in either of the components of σ, but not in both components, formally $cov(\sigma_L) \cap cov(\sigma_R) = \emptyset$. For the same reason, the leaves are also *fully covered* either in σ_L or σ_R, so $pc(\sigma_L) = pc(\sigma_R) = \emptyset$. The same is true for all descendants σ' of σ that are in $\Sigma^{>1L}$, which do not exactly cover any leaf. Thus, $\mathcal{N}(\sigma') \leq 1$ for all $\sigma' \in \Sigma^{>1L}$, concluding the proof. □

Additional Experiments

In the main paper, we indicate that the number of state evaluations differs only for very few instances when the FM is not forced to be general. We show detailed results about this in Figure 2 (right). When there is a difference in state evaluations (the points not on the diagonal), this difference is almost always smaller than a factor of two. There are only five instances where the factor is larger, with a maximum of 16. In comparison with the runtime plot (Figure 2, left), though, it is clear that the differences in runtime are mostly independent from the state evaluations.

References

Edelkamp, S.; and Kissmann, P. 2008. Limits and Possibilities of BDDs in State Space Search. In *Proceedings of*

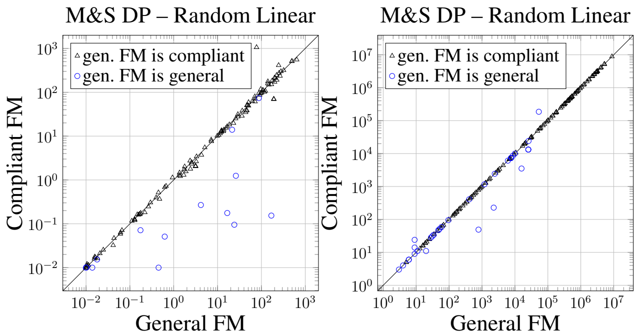

Figure 2: Per-instance comparison of the search time (left) and the number of state evaluations (right) for M&S with random linear merging comparing general to compliant FMs.

the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI 2008), 1452–1453. AAAI Press.

Edelkamp, S.; Kissmann, P.; and Torralba, Á. 2012. Symbolic A* Search with Pattern Databases and the Merge-and-Shrink Abstraction. In De Raedt, L.; Bessiere, C.; Dubois, D.; Doherty, P.; Frasconi, P.; Heintz, F.; and Lucas, P., eds., *Proceedings of the 20th European Conference on Artificial Intelligence (ECAI 2012)*, 306–311. IOS Press.

Torralba, Á. 2015. *Symbolic Search and Abstraction Heuristics for Cost-Optimal Planning*. Ph.D. thesis, Universidad Carlos III de Madrid.