Custom-Design of FDR Encodings: The Case of Red-Black Planning

Submission 2

Abstract
Classical planning tasks are commonly described in PDDL, while most planning systems operate on a grounded finite-domain representation (FDR). The translation of PDDL into FDR is complex and has a lot of choice points—it involves identifying so-called mutex groups—but most systems rely on the translator that comes with Fast Downward. Yet the translation choice points can strongly impact performance. Prior work has considered optimizing FDR encodings in terms of the number of variables produced. Here we go one step further by proposing to custom-design FDR encodings, optimizing the encoding to suit particular planning techniques. We develop such a custom design here for red-black planning, a partial delete relaxation technique. The FDR encoding affects the causal graph and the domain transition graph structures, which govern the tractable fragment of red-black planning and hence affects the respective heuristic function. We develop integer linear programming techniques optimizing the scope of that fragment in the resulting FDR encoding. We empirically show that the performance of red-black planning can be improved through such FDR custom design.

Introduction
Classical planning tasks are usually defined in PDDL (McDermott 2000), which is a lifted representation based on first-order logic. Yet, most state-of-the-art planners, in particular the systems based on Fast Downward (FD) (Helmert 2006), use a finite-domain representation (FDR) encoding, where subsets of propositional facts are grouped into finite-domain state variables. This is possible only if reachable states make at most one of the grouped facts true, i.e., if they form a mutex group. FDR encodings are useful for a wide variety of purposes like abstraction based heuristics (Haslum et al. 2007; Katz and Domshlak 2010; Helmert et al. 2014; Seipp and Helmert 2018), or red-black planning heuristics (Domshlak, Hoffmann, and Katz 2015), where a subset of (red) variables are delete-relaxed while the rest (black) keep their true semantics.

The translation of PDDL into FDR is complex—deciding whether a set of facts forms a mutex group is as hard as planning itself. There are many choice points in identifying mutex groups and grouping facts into state variables. Systems based on Fast Downward almost universally rely on its translator (Helmert 2009) and do not question the choices made there.

Yet the translation choices can impact performance. In prior work (Dvořák, Toropila, and Barták 2013, 2015), a minimization of the number of FDR variables was proposed assuming fewer variables result in more information being aggregated into them. The results show that significantly reducing the number of variables can improve the performance of FDR-based techniques. The improvements are moderate though, and there is a lot of variance depending on domain and planning technique; a consistent picture does not emerge.

Here we show that this picture can change when going one step further in the target of FDR optimization: we propose to custom-design FDR encodings, optimizing the encoding to suit particular planning techniques. 1 We develop such a custom design for red-black planning, which distinguishes between relaxed (red) and non-relaxed (black) variables. We focus on a polynomial-time solvable fragment of red-black planning (Katz, Hoffmann, and Domshlak 2013b) which uses a heuristic function based on generating a red-black plan for a given state (Katz, Hoffmann, and Domshlak 2013a). The tractability of red-black planning is determined by the decision which variables are painted red or black. The state-of-the-art approach uses painting strategies (Domshlak, Hoffmann, and Katz 2015) assuming the FDR encoding is given and fixed, and decides only on variable colors. Here, we open up this design space and allow to custom-design the FDR variables and their painting at the same time.

The tractability of red-black planning is governed by the FDR encoding’s causal graph (whose projection onto the black variables must be acyclic) and domain transition graphs (that must be invertible for the black variables). We develop techniques optimizing these explicit syntactic criteria, to be able to choose from different viable FDR encodings of tractable red-black planning tasks. The optimization itself is then formulated as an integer linear program, where the allocation of black (non-relaxed) variables is controlled.

1Vallati et al. (2015; 2017) have explored domain model optimization tailored to specific planners before, but for PDDL not for FDR variable design, and changing only the ordering of artefacts in the input model (which had previously been observed to be relevant for the performance of some planners (Howe and Dahlman 2002)).
by the program’s objective function.

Background

A STRIPS planning task $\Pi$ is specified by a tuple $\Pi = (F, O, s_i, s_G),$ where $F = \{f_1, \ldots, f_n\}$ is a set of facts, and $O = \{o_1, \ldots, o_m\}$ is a set of grounded operators. A state $s \subseteq F$ is a set of facts, $s_i \subseteq F$ is an initial state and $s_G \subseteq F$ is a goal specification. A state $s$ is a goal state iff $s \subseteq s_G$. An operator $o$ is a tuple $o = (\text{pre}(o), \text{add}(o), \text{del}(o))$, where pre$(o)$ is the set of preconditions of $o$, and add$(o)$ and del$(o)$ are the sets of add and delete effects, respectively. All operators are well-formed, i.e., add$(o) \cap \text{del}(o) = \emptyset$ and pre$(o) \cap \text{del}(o) = \emptyset$. We assume unit costs. An operator $o$ is applicable in a state $s$ if pre$(o) \subseteq s$. The resulting state of applying an applicable operator $o$ in a state $s$ is the state $o[s] = (s \setminus \text{del}(o)) \cup \text{add}(o)$.

A sequence of operators $\pi = \langle o_1, \ldots, o_n \rangle$ is applicable in a state $s_0$ if there are states $s_1, \ldots, s_n$ such that $o_i$ is applicable in $s_i-1$ and $s_i = o_i[s_{i-1}]$ for $1 \leq i \leq n$. The resulting state of this application is $\pi[s_0] = s_n$. A sequence of operators $\pi$ is called a plan if $\pi$ is applicable in $s_I$ and $s_G \subseteq \pi[s_I]$.

A set of facts $M \subseteq F$ is called a mutex if $M \not\subseteq s$ for every reachable state $s$. A set of facts $M \subseteq F$ is called a mutex group if $|M \cap s| \leq 1$ for every reachable state $s$, and $M$ is called a fact-alternating mutex group (fam-group) if $|M \cap s| \leq 1$ and $|M \cap \text{add}(o)| \leq |M \cap \text{pre}(o) \cap \text{del}(o)|$ for every operator $o \in O$. Every fam-group is a mutex group. Every subset of a mutex group is a mutex group, but not every subset of a fam-group is a fam-group. Facts from every mutex group form pairwise mutexes (Fišer and Komenda 2018).

Given a set of mutex groups $G$, $M_G \subseteq 2^F$ denotes an upper set of mutexes induced by $G$, i.e., for every mutex group $G \in G$ and every $f, f' \in G$, $f \neq f'$, it holds that $\{f, f'\} \in M_G$, and for every $M \in M_G$ and every $f \in F$ it holds that $M \cup \{f\} \in M_G$.

An FDR planning task $\mathcal{P}$ is specified by a tuple $\mathcal{P} = (V, \psi_I, \psi_G)$. $V$ is a finite set of variables, each variable $v \in V$ has a finite domain $\text{dom}(V)$. A fact $(V, v)$ is a pair of a variable $v \in V$ and one of its values $v \in \text{dom}(V)$. A partial state $p$ is a variable assignment over some variables $\text{vars}(p) \subseteq V$. We write $p[V]$ for the value assigned to the variable $v \in \text{vars}(p)$ in the partial state $p$. Given a set of variables $U \subseteq V$, $p[U]$ denotes a partial state $p$ restricted to $U$. We also identify $p$ with the set of facts contained in $p$, i.e., $p = \{v \in V \mid v \in \text{vars}(p)\}$. A partial state $s$ is a state if $\text{vars}(s) = V$, $\psi_I$ is an initial state. $\psi_G$ is a partial state called goal, and a state $s$ is a goal state iff $\psi_G \subseteq s$. $\mathcal{P}$ is a finite set of operators, each operator $o \in \mathcal{P}$ has a precondition pre$(o)$ and effect eff$(o)$, which are partial states. An operator $o$ is applicable in a state $s$ iff pre$(o) \subseteq s$. The resulting state of applying an applicable operator $o$ in a state $s$ is the state $o[s]$ where $o[s][v] = \text{eff}(o)[v]$ for every $v \in \text{vars}(\text{eff}(o))$, and $o[s][v] = s[v]$ for every $v \in V \setminus \text{vars}(\text{eff}(o))$. Operator sequences and plans are defined analogously to STRIPS.

The domain transition graph (DTG) of a variable $V$, denoted by $\mathcal{D}_V$, is an edge-labeled multi-diagram with vertices $\text{dom}(V)$ and with an edge from $d$ to $d'$ induced by an operator $o \in \mathcal{P}$ and denoted by $(d, q, d')$ iff $d \neq d'$, $\text{eff}(o)[V] = d'$, and either $\text{pre}(q)[V] = d$ or $V \notin \text{vars}(\text{pre}(q))$.

Let $U \subseteq V$ denote a subset of variables. The causal graph $\mathcal{C}_G$ of $U$ is a digraph with vertices $U$. An edge $(V, V')$ is in $\mathcal{C}_G$ iff $V \neq V'$ and there exists an operator $o \in \mathcal{P}$ such that $(V, V') \in (\text{vars}(\text{pre}(o)) \cup \text{vars}(\text{eff}(o))) \times \text{vars}(\text{eff}(o))$.

A red-black planning task $\Pi$ is specified by a tuple $\Pi = (\mathcal{V}^R, \mathcal{V}^B, \mathcal{O}, \psi_I, \psi_G)$, where $\mathcal{V}^R$ and $\mathcal{V}^B$ are state variables, called red variables and black variables, respectively. Black variables have the value-switching semantics as in FDR. Red variables have the value-accumulating semantics, i.e., operators “extend” the value of $v \in \mathcal{V}^R$ from $\{x, y\}$ to $\{x\}$. Applicability of operators, sequences of operators, and red-black plans are defined accordingly (see Domshlak, Hoffmann, and Katz 2015).

Given a variable $V \in \mathcal{V}$, an edge $(d, q, d')$ from $\mathcal{D}_V$ is relaxed side effects invertible (RSE-invertible) if there exists an edge $(d', q', d)$ such that $\text{pre}(q')[V \setminus V] \subseteq \text{pre}(q)[V \setminus V] \cup \text{eff}(q)[V \setminus V]$. $V$ is RSE-invertible if every edge in $\mathcal{D}_V$ is RSE-invertible.

The red-black heuristic $h_{\text{RB}}$ for a state $s$ is defined as the length of a red-black plan for $s$. Here, we focus on tractable fragments of red-black planning. One such fragment is characterized by RSE-invertible black variables $\mathcal{V}^B$ that form an acyclic causal graph $\mathcal{C}_{G^B}$ (Domshlak, Hoffmann, and Katz 2015).

Translation from STRIPS to FDR

A concise translation from STRIPS to FDR requires mutex groups that are used for creating FDR variables from STRIPS facts. Before we formalize the translation process, we formulate an intermediate STRIPS representation where we assume the mutex groups are already inferred and the STRIPS planning task is pruned with the mutex information contained in them. Moreover, we need to preserve the DTG structure of black variables to preserve their RSE-invertibility. We introduce a mapping from mutex groups to fam-groups containing these mutex groups. We choose specifically fam-groups because of their simpler structure (Fišer and Komenda 2018). It is not a limitation, since fam-groups are the most commonly used type of mutex groups anyway (Fišer 2020).

Definition 1. Given a set of mutex groups $G$, $\gamma : G \rightarrow 2^F$ is a fam-group map if for every $G \in G$ it holds that either $\gamma(G) = \emptyset$ or $\gamma(G)$ is a fam-group such that $G \subseteq \gamma(G)$.

Definition 2. Given a STRIPS planning task $\Pi = (F, O, s_i, s_G)$, and a set of non-empty mutex groups $G$, and a fam-group map $\gamma$ over $G$, $E_{\Pi} = \langle \Pi, G, \gamma \rangle$ denotes a STRIPS planning task extended with mutex groups (MGE-STRIPS) if all of the following hold:

(S1) $F = \bigcup_{o \in O} \text{add}(o) \cup \text{del}(o)$.

(S2) for every operator $o \in O$ it holds that $\text{pre}(o) \notin M_G$ and $o[\text{pre}(o)] \notin M_G$ and for every $f \in \text{del}(o)$ it holds that $\{f\} \cup \text{pre}(o) \notin M_G$. 

Although encoding $\perp_G$ in preconditions of operators is not necessary, because this fact is true implicitly, we use it to ensure that the structure of a projection to the mutex group in STRIPS is preserved in the corresponding DTG in FDR. Why we need this will become clear in the next section where we deal with RSE-invertibility. For now, see the example depicted in Figure 1. The projection to the fam-group $F = \{f_1, \ldots, f_4\}$ is shown in Figure 1a. Now suppose we decide to encode only a subset of $F$ as an FDR variable, namely $G = \{f_1, f_2\}$. If we set $\gamma(G) = 0$, then the resulting DTG of $V_G$ will be as depicted in Figure 1c, i.e., it will have additional edges between $f_1$ and $f_2$ not present in the original projection, because the operators $o_{31}$ and $o_{42}$ have $f_3$ and $f_4$, respectively, in their preconditions. Therefore $V_G$ is not set in $\text{pre}(q_{o_{31}})$ and $\text{pre}(q_{o_{42}})$. However, if we set $\gamma(G) = 0$, then we preserve the structure, as depicted in Figure 1b, because $V_G$ will be set to $\perp_G$ in the preconditions of $q_{o_{31}}$ and $q_{o_{42}}$.

**Proposition 4.** Let $E_\Pi = \langle \Pi = \langle F, O, s_I, s_G, G, \gamma \rangle, \mathcal{P}(E_\Pi) = \langle V, O, \psi_I, \psi_G \rangle \rangle$ denote an MGE-STRIPS planning task. $\pi = \langle o_1, \ldots, o_n \rangle$ is a plan for $E_\Pi$ iff $\pi' = \langle q_{o_1}, \ldots, q_{o_n} \rangle$ is a plan for $\mathcal{P}(E_\Pi)$.

**Proof Sketch.** $\mathcal{P}(E_\Pi)$ is well-defined, because (T3) assigns one value to every variable, i.e., $\psi_I$ is a state. Similarly, (T4) ensures that $\psi_G$ is a partial state, and (T2a-b) produce partial states, because only one value is assigned to each variable.

Note that $f \in s_I$ iff $\langle V_G, f \rangle \in \psi_I$ for every $G$ such that $G \cap s_I = \{f\}$, and $\langle V_G, \perp_G \rangle \in \psi_I$ only if $s_I \in M_G$. Similarly, $f \in s_I$ iff $\langle V_G, f \rangle \in \psi_G$; and for every $o \in O$ it holds that $f \in \text{pre}(o) \text{ iff } \langle V_G, f \rangle \in \text{pre}(o)$, and $\langle V_G, \perp_G \rangle \in \text{pre}(o)$ only if $o \in M_G$. Hence $o$ is applicable in $s_I$ iff $q_{o} \in \psi_I$ for every $o \in O$. Furthermore, from (T2b) it follows that $f \in \text{add}(s_I) \text{ iff } \langle V_G, f \rangle \in q_{o} \psi_I$ for every $G$ such that $G \cap o(s_I) = \{f\}$, and $\langle V_G, \perp_G \rangle \in q_{o} \psi_I$ only if $G$ is mutex with $o(s_I)$. The rest follows by induction.

**RSE-Invertible Variables**

Now, we show how to identify mutex groups in $E_\Pi$ whose counterparts form variables in $\mathcal{P}(E_\Pi)$ that are RSE-invertible. This condition is required for black variables in the tractable fragment of red-black planning we consider. First, we formulate conditions for RSE-invertibility of facts and mutex groups. Then, we show that, given a fam-group map, these conditions are both sufficient and necessary for RSE-invertibility of the resulting FDR variables.

For the rest of this section, let $E_\Pi = \langle \Pi = \langle F, O, s_I, s_G, G, \gamma \rangle, \mathcal{P}(E_\Pi) = \langle V, O, \psi_I, \psi_G \rangle \rangle$ denote an MGE-STRIPS planning task, and let $\mathcal{P}(E_\Pi) = \langle V, O, \psi_I, \psi_G \rangle$ denote an FDR planning task constructed from $E_\Pi$ according to Definition 3.

**Definition 5.** A fact $f \in F$ is RSE-invertible if

(1) there exist $o, o' \in O$ such that $f \in \text{add}(o)$ and $f \in \text{del}(o')$, and

(2) for every operator $o \in O$ such that $f \in \text{del}(o)$ it holds that there exists an operator $o' \in O$ such that $f \in \text{add}(o')$ and $\text{pre}(o') \subseteq \text{pre}(o) \cup \text{add}(o)$, and
(13) for every operator \( o \in \mathcal{O} \) such that \( f \in \text{add}(o) \) it holds that there exists an operator \( o' \in \mathcal{O} \) such that \( f \in \text{del}(o') \) and \( \text{pre}(o') \subseteq \text{pre}(o) \cup \text{add}(o) \).

We also call a set of facts (or mutex group) \( F \subseteq F \) RSE-invertible if every \( f \in F \) is RSE-invertible.

For the main result of this section (Theorem 9), we need to prove several auxiliary lemmas first. We start with showing that the condition \( \text{pre}(o) \subseteq \text{pre}(o) \cup \text{add}(o) \) from (12-3) corresponds to the RSE-invertibility condition on edges in DTGs as long as we have transitions in both directions.

**Lemma 6.** Let \( G \in \mathcal{G} \) be a mutex group and \( o', o \in \mathcal{O} \) be two operators with edges \((x, q_o, x')\) and \((x', q_{o'}, x)\) in the DTG of \( V_G \). Then \( \text{pre}(o) \subseteq \text{pre}(o') \cup \text{add}(o') \) if \( \text{pre}(q_o) \subseteq \text{pre}(q_{o'}) \). Therefore \( \text{pre}(o) \subseteq \text{pre}(o') \cup \text{add}(o') \).

**Proof.** \( \Rightarrow \): It is easy to see that \( f \in \text{pre}(o) \) iff \( (V_G, f) \in \text{pre}(q_o) \) and \( f \in \text{pre}(o') \) iff \( (V_G, f) \in \text{pre}(q_{o'}) \), therefore \( \text{pre}(o) \subseteq \text{pre}(o') \cup \text{add}(o') \).

\( \Leftarrow \): For every \( G \in \mathcal{G} \) s.t. \( G \cap \text{pre}(o) \neq \emptyset \), there exists \( f \in \mathcal{G} \) s.t. \( f \in \text{pre}(o) \) and \( \text{pre}(q_o) \subseteq \text{pre}(q_{o'}) \). Therefore \( \text{pre}(o) \subseteq \text{pre}(o') \cup \text{add}(o') \).

A mutex group \( M \) with \( |M| \geq 2 \) requires the fam-group map \( \gamma \) to map \( M \) to its superset fam-group. Recall the example in Figure 1. The facts \( f_1 \) and \( f_2 \) are RSE-invertible, but \( f_3 \) and \( f_4 \) are not, because of the operator \( o_3 \). So, we would like to construct an RSE-invertible variable from \( G = \{ f_1, f_2 \} \). If we set \( \gamma(G) = \emptyset \), the resulting DTG (Figure 1c) will have edges between \( f_1 \) and \( f_2 \) that are not RSE-invertible, because the operator \( o_3 \) has \( f_3 \) as its pre-condition, which is not part of \( \text{pre}(o_{32}) \cup \text{add}(o_{32}) \). However, \( \gamma(G) = F \) results in the RSE-invertible DTG (Figure 1b).

**Lemma 7.** Let \( G = \{ f \} \) be a mutex group. Then \( f \) is RSE-invertible iff \( V_G \) is RSE-invertible.

**Proof.** \( \Rightarrow \): Since \( \text{dom}(V_G) = \{ f, \bot_G \} \) and we need to prove that every edge in the DTG of \( V_G \) is RSE-invertible, we need to investigate two cases:

1. Let \((f, q_o, \bot_G)\) be an edge in the DTG of \( V_G \). Then \( \text{eff}(q_o)[V] = \bot_G \) and \( \text{pre}(q_o)[V] = f \) or \( V \notin \text{vars}(\text{pre}(q_o)) \). So we have \( f \notin \text{add}(o) \) and \( f \in \text{del}(o) \), therefore there exists \( o' \) s.t. \( f \in \text{add}(o') \) and \( \text{pre}(o') \subseteq \text{pre}(o) \cup \text{add}(o) \). So, \( f \notin \text{pre}(o') \) because \( f \in \text{add}(o') \), therefore \( \text{pre}(q_o)[V_G] = \bot_G \) or \( V_G \notin \text{vars}(\text{pre}(q_o)) \), therefore there is an edge \((\bot_G, q_{o'}, f)\).

2. (2a) If \((f, q_o, \bot_G)\) be an edge in the DTG of \( V_G \). Then \( \text{eff}(q_o)[V_G] = f \) and either \( \text{pre}(q_o)[V_G] = \bot_G \) or \( V \notin \text{vars}(\text{pre}(q_o)) \). So, we have \( f \in \text{add}(o) \) and therefore \( f \notin \text{pre}(o) \) and \( f \notin \text{del}(o) \). Therefore, there exists \( o' \in \mathcal{O} \) s.t. \( f \in \text{del}(o') \) and \( \text{pre}(o') \subseteq \text{pre}(o) \cup \text{add}(o) \). Now, to prove that there is an edge \((f, q_{o'}, \bot)\) (and therefore this lemma follows from Lemma 6), we need to show that \( \text{pre}(q_o)[V_G] \neq \bot_G \), i.e., \( \text{pre}(q_o)[V_G] = f \) or \( V \notin \text{vars}(\text{pre}(q_o)) \). If \( \text{pre}(q_o)[V_G] = \bot_G \), then there is \( f' \in \gamma(G) \) s.t. \( f' \in \text{pre}(o') \cap \text{del}(o') \), because \( \gamma(G) \) is a fam-group. Therefore, \( \{f, f'\} \subseteq \text{del}(o') \), which is a contradiction because \( \{f\} \cup \text{pre}(o') \) is a mutex (S2).

\( \Leftarrow \): If \( V_G \) is RSE-invertible, then for every edge \((x, q_o, x')\) in \( D_{V_G} \) there exists \((x', q_{o'}, x)\) such that \( \text{pre}(q_o)[V \setminus V_G] \subseteq \text{pre}(q_{o'})[V \setminus V_G] \cup \text{eff}(q_o)[V \setminus V_G] \). Therefore, it follows from Lemma 6 that also \( \text{pre}(o') \subseteq \text{pre}(o) \cup \text{add}(o) \). Moreover, from the construction (T2b) it follows that if \( x' = f \), then \( f \in \text{add}(o) \) and \( f \in \text{del}(o) \), and vice versa, if \( x = f \), then \( f \in \text{del}(o) \) and \( f \in \text{add}(o) \). Therefore, the conditions (11-3) are satisfied.

Now, we are ready to prove that RSE-invertibility of mutex groups implies RSE-invertibility of the corresponding FDR variables, and vice versa. We separate the case of a singleton mutex group, because it turns out that larger mutex groups require an additional condition on a fam-group map.

**Lemma 8.** Let \( G \in \mathcal{G} \) be a mutex group with \( |G| \geq 2 \) and \( \gamma(G) \neq \emptyset \). Then \( G \) is RSE-invertible iff \( V_G \) is RSE-invertible.

**Proof.** \( \Rightarrow \): Since \( \gamma(G) \) is a fam-group s.t. \( G \subseteq \gamma(G) \), for every operator \( o \in \mathcal{O} \) and every \( f \in G \) it holds that \( f \in \text{del}(o) \) implies \( f \in \text{pre}(o) \). Let \((d, q_o, d')\) denote an edge in the DTG of \( V_G \). Then \( \text{eff}(q_o)[V_G] = d' \) and \( \text{pre}(q_o)[V_G] = d \) or \( V_G \notin \text{vars}(\text{pre}(q_o)) \). Now, if we show that there also exist an edge \((d', q_{o'}, d)\) for some \( o' \in \mathcal{O} \) s.t. \( \text{pre}(o') \subseteq \text{pre}(o) \cup \text{add}(o) \), then the rest follows from Lemma 6. So three cases need to be investigated.

1. If \( d, d' \in G \), then \( d' \in \text{add}(o) \). And since \( \gamma(G) \) is a fam-group, we have \( V_G \subseteq \text{vars}(\text{pre}(q_o)) \) and \( d \in \text{pre}(o) \cap \text{del}(o) \). Therefore, there exists \( o' \in \mathcal{O} \) s.t. \( d \in \text{add}(o') \) and \( \text{pre}(o') \subseteq \text{pre}(o) \cup \text{add}(o) \). Thus, \( d' \in \text{pre}(o') \), because \( d \notin \text{pre}(o') \) and \( \gamma(G) \cap (\text{pre}(o) \cup \text{add}(o)) = \{d, d'\} \). Therefore, there exists an edge \((d', q_{o'}, d)\).

2. If \( d \in G \) and \( d' = \bot_G \), then \( \text{add}(o) \cap \text{G} = \emptyset \) and \( \text{del}(o) \cap \text{G} = \emptyset \), therefore \( d \in \text{del}(o) \), because otherwise there would exist \( x \in G \), \( x \neq d, x \in \text{del}(o) \), therefore \( x \in \text{pre}(o) \), therefore \( \text{pre}(q_o)[V_G] = x \) which is in contradiction. So we have \( d \in \text{del}(o) \) and also \( d \in \text{pre}(o) \),
Therefore, there exists $o'$ s.t. $d \in add(o')$ and $pre(o') \subseteq pre(o) \cup add(o)$. So since $d \in pre(o)$ and $add(o) \cap G = \emptyset$, then $pre(o') \cap G = \emptyset$, therefore we have $eff(q_o)[V_G] = d$ and either $pre(q_o)[V_G] = \bot_G$ or $V_G \not\in \text{vars}(pre(q_o))$.

Therefore, we have an edge $(\bot_G, q_o, d)$.

(3) If $d = \bot_G$ and $d' \in G$, then $d' \in add(o)$, therefore there exists $f \in (G) \setminus G$ s.t. $f \in pre(o) \cap del(o)$.

Thus, there exists $o' \in O$ s.t. $f \in add(o')$ and $pre(o') \subseteq pre(o) \cup add(o)$. Hence, $d' \in pre(d') \cap del(o)$ because $\gamma(G) \cap pre(d') \cap del(o) = \emptyset$ ($\gamma(G)$ is a fam-group) and $(pre(o) \cup add(o)) \cap (\gamma(G)) = \{f, d', f \not\in \gamma(G)\}$. Therefore, $pre(q_o)[V_G] = d'$ and $eff(q_o)[V_G] = \bot_G$, and thus we have $(d', q_o, \bot_G)$.

“⇐”: Let $(x, q_o, x')$ denote an edge in $\mathcal{D}_{V_G}$. If $x' \in G$, then $x' \in add(o)$ (T2b). Since $\gamma(G)$ is a fam-group, either $pre(q_o)[V_G] = \bot_G$, or $pre(q_o)[V_G] = x$ s.t. $x \in G$. Therefore if $x \in G$, then $x \in pre(o) \cap del(o)$. Therefore, for every such $q_o$ it holds that either $G \cap del(o) \neq \emptyset$ or $G \cap add(o) \neq \emptyset$.

Since $V_G$ is RSE-invertible, for every edge $(x, q_o, x')$ there exist an edge $(x', q_o, x)$ s.t. $pre(q_o)[V_G] \subseteq pre(q_o)[\mathcal{V}_G] \cup eff(q_o)[V_G] \cup \mathcal{V}_{F_G}$, therefore it follows from Lemma 6 that $pre(o') \subseteq pre(o) \cup \text{add}(o)$. Hence, the conditions (II-3) are satisfied.

Theorem 9. Let $G \in G$ be a mutex group with $|G| = 1$ or $\gamma(G) \neq \emptyset$. Then $G$ is RSE-invertible iff $V_G$ is RSE-invertible.

Proof. It follows directly from Lemma 7 and Lemma 8.

Acyclic Causal Graph

Besides RSE-invertibility of black variables, the tractable fragment of red-black planning requires the black causal graph to be acyclic. To achieve that, we show how to find a causal link between mutex groups that is translated into FDR.

Definition 10. Given a mutex group $G \in 2^F$, two distinct facts $f \in G$ and $f' \not\in G$, and a fam-group map $\gamma$ on $2^F$, we say that there is a causal link from $(G, f)$ to $f'$ iff there exists an operator $o \in O$ such that (i) $f' \notin \text{del}(o) \cup \text{add}(o)$, and (ii) $f \in pre(o) \cap \text{del}(o) \cup \text{add}(o)$, or $\gamma(G) \cap pre(o) \neq \emptyset$.

Theorem 11. Let $E_{II} = (\Pi, G, \gamma)$ be an MGE-STRIPS task with operators $O$, $A, B \in G$ be two distinct mutex groups, and $CG_V$ be the causal graph of $G_{E_{II}}$. There is an edge $(V_A, V_B)$ in $CG_V$ if there exist $f, f' \in F$ such that there is a causal link from $(A, f)$ to $f'$.

Proof. “⇐”: Let $o \in O$ be the operator inducing the causal link from $(A, f)$ to $f'$. If $f \in pre(o)$, then $pre(q_o)[V_A] = f$. If $f \in add(o)$, then $eff(q_o)[V_A] = f$. If $f \in del(o)$, then either $eff(q_o)[V_A] = \bot_A$ or there exist $x \in A$ s.t. $eff(q_o)[V_A] = \bot_A$. If $\gamma(A) \neq \emptyset$ and $\gamma(A) \cap pre(o) \neq \emptyset$, then either $pre(q_o)[V_A] = f$ for some $x \in A$, or $pre(q_o)[V_A] = \bot_A$. Therefore, $V_A \in \{\text{vars}(pre(q_o)) \cup \text{vars}(eff(q_o))\}$, and for similar reasons $V_B \in \text{vars}(eff(q_o))$.

“⇒”: If $(V_A, V_B)$ in $CG_{\{V_A, V_B\}}$, then there exists $q_o \in O$ s.t. $V_A \in \{\text{vars}(pre(q_o)) \cup \text{vars}(eff(q_o))\}$ and $V_B \in \text{vars}(eff(q_o))$. If $pre(q_o)[V_A] \in A$ or $eff(q_o)[V_A] \in A$, and $eff(q_o)[V_B] \in B$, then it follows trivially that $A \cap \text{pre}(o) \cup \text{add}(o) \neq \emptyset$ and $B \cap \text{add}(o) \neq \emptyset$. So what remains is to show that (i) if $pre(q_o)[V_A] = \bot_A$, then $\gamma(A) \neq \emptyset$, and if $\gamma(A) \cap \text{pre}(o) \neq \emptyset$ (T2a); (ii) if $eff(q_o)[V_A] = \bot_A$, then $A \cap \text{del}(o) \neq \emptyset$ (T2b); and (iii) if $eff(q_o)[V_B] = \bot_B$, then $B \cap \text{del}(o) \neq \emptyset$ (T2b).

Inference of Black Variables

We have shown how to find RSE-invertible mutex groups and what conditions must be met in order to form a causal link between FDR variables created from mutex groups. Now, we put everything together and show how to select mutex groups that can be translated into RSE-invertible FDR variables that form an acyclic causal graph, and thus can be painted black.

Definition 12. Given a STRIPS planning task $II$ with facts $F$, a fam-group map $\gamma$ over $2^F$, and a set of mutex groups $K \subseteq 2^F$, such that for every $G \in K$ it holds that either $|G| = 1$, or $\gamma(G) \neq \emptyset$. $X_{\Pi, K, \gamma}$ denotes a digraph with vertices $V \subseteq K \times F$ and edges $E \subseteq V \times V$, where $(G, f) \in V$ iff $G \in K$ and $f \in G$, and $((G, f), (G', f')) \in E$ iff $f = f'$ or there is a causal link from $(G, f)$ to $f'$.

Given a cycle $c$ in $\Pi, K, \gamma$, let $C(c) = \{G \mid (G, f) \in c\}$.

Theorem 13. Let $E_{II} = (\Pi, G, \gamma)$ be an MGE-STRIPS planning task, $\Pi \subseteq G$ be a set of RSE-invertible mutex groups with $|G| = 1$ or $\gamma(G) \neq \emptyset$, and $C_G$ be the set of corresponding variables from $G_{E_{II}}$. Then $CG_{V_G}$ is acyclic iff for every cycle $c$ in $\Pi, K, \gamma$ it holds that $|C(c)| = 1$.

Proof Sketch. It follows directly from Theorem 11, because if $|C(c)| = 1$, then every cycle of causal links is contained within one mutex group.

Algorithm 1 encapsulates the algorithm for inference of a set of RSE-invertible mutex groups $K$, and a fam-group map $\gamma$, that can be translated into red-black planning task as black variables. We assume we are given a STRIPS planning task $II$, and a set of fam-groups $H \subseteq 2^F$ that can be inferred by one of the algorithm proposed by Helmert (2009),
Fišer and Komenda (2018), or Fišer (2020). In steps 1-3, we prepare RSE-invertible mutex groups and a mapping to the corresponding fam-groups. In step 4, the graph defined in Definition 12 is constructed, and in step 5, we find a subset of vertices having cycles only within each mutex group. Finally, in the last three steps we extract the solution.

Note that the step 5 is a variant of an NP-complete feedback vertex set problem (Karp 1972). We decided to use a naive approach and solve this problem by an integer linear program (ILP), where each vertex from $X(\Pi, H^+, \gamma')$ corresponds to a binary variable, and each constraint corresponds to a cycle that has to be avoided. Instead of listing all cycles, we first add all cycles between pairs of vertices from different mutex groups, then we solve the problem, and check whether the resulting solution has a cycle. If we find a cycle, we add it as another constraint, solve the problem, and continue in this manner until we find a feasible solution.

As objective function of the ILP, we tried two variants. The first variant is simply a maximization of the number of facts in the resulting mutex groups. The second variant weights facts using the number of conflicts in a relaxed plan $\pi^+$ obtained from the FF heuristic (Hoffmann and Nebel 2001) computed on the STRIPS representation. Following prior work (Domshlak, Hoffmann, and Katz 2015), we say a fact $f$ has a conflict if there is an operator $o$ in $\pi^+$ such that $f \in \text{pre}(o)$ and $f$ is not satisfied when executing $\pi^+$ with the non-relaxed semantics. For each mutex group $M$, we sum the number of conflicts from all $f \in M$, and we set the weight of each fact $f \in M$ as the sum over $|M|$. This way we prioritize mutex groups containing facts with most conflicts in the relaxed plan.

Experimental Evaluation

We implemented Algorithm 1 in C, used fam-groups inferred from PDDL (Fišer 2020), and used the red-black heuristic $h_{\text{RB}}$ (Domshlak, Hoffmann, and Katz 2015) implemented in FD (Helmert 2006). The experiments were performed on Intel(R) Xeon(R) Scalable Gold 6146 machines, with time and memory limits of 30min and 8GB. Out of all benchmarks from International Planning Competitions of 1998 to 2018 satisfying planning tracks, we selected all tasks where no conditional effects were created and at least one variable could be painted black.\footnote{We will make our code and experimental data public.}

We compare the following configurations: the baseline configuration B runs the painting strategy denoted as $A^3$ by Domshlak, Hoffmann, and Katz (2015) on the default FDR encoding (Helmert 2009); C is the variant of Algorithm 1 using conflicts in a relaxed plan; and the “oracle” O picks the best FDR encoding and painting for each tested task. It considers B, and five best encodings (by objective value) from each of M and C. The encodings are chosen in terms of coverage, preferring tasks solved in the initial state by $h_{\text{RB}}$, 0 serves to illustrate the potential of our customized encodings, assuming perfect knowledge about which encoding works best for a given task.

The scatter plots in Figure 2 show that our methods are able to increase the number of black facts by several orders of magnitude. The oracle often uses significantly less black facts than B, indicating that it is not necessarily beneficial to have more black facts. In fact, having more black variables (facts) can negatively impact heuristic computation time, without improving informativeness.

Table 1 shows the number of tasks solved (coverage) in the left, and the number of tasks solved by $h_{\text{RB}}$ in the initial state in the right. C clearly outperforms B which, again, shows that maximization of the number of black facts is not necessarily the best painting strategy. C and B are very similar in coverage, with strengths in different domains. C can solve 5 more tasks in the initial state resulting from only two domains, blocks00 and satellite02.

The oracle variant $O$ demonstrates the potential of our approach if we better understand how to select the best encoding for a given task. B already solves a large portion of tasks,
but here we show that it is still possible to create an encoding and painting that could further increase this number by at least 25 tasks, and 14 more can be solved in the initial state. Unfortunately, it is unclear how to directly compare the quality of individual encodings for a given task without running the search. Maximizing the number of black facts turned out to be too simplistic as an optimization criterion. Considering conflicts in a relaxed plan improves the results in some domains, but it is still unclear what properties identify a good FDR encoding and painting for red-black planning.

**Conclusion**

We devised a method to custom-design FDR encodings of classical planning tasks, showcasing how a tractable fragment of red-black planning can be formulated into the translation from PDDL to FDR. By describing this translation as an integer linear program, we can explore the space of possible tractable red-black FDR encodings. Our evaluation shows an increase in the number of black facts by several orders of magnitude. Our approach is flexible, allowing to express complex properties based on, e.g., DTGs or causal graphs. Thus, custom-designed FDR encodings have the potential of benefiting planning techniques beyond red-black planning, opening up a wide range of possibilities for future research.

**References**


