LM-Cut Heuristics for Optimal Linear Numeric Planning

Ryo Kuroiwa[1], Alexander Shleyfman[2], J. Christopher Beck[1]

[1]Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Canada, ON M5S 3G8
[2]Technion, Haifa, Israel
ryo.kuroiwa@mail.utoronto.ca, shleyfman.alexander@gmail.com, jcb@mie.utoronto.ca

Abstract

While numeric variables play an important, sometimes central, role in many planning problems arising from real world scenarios, most of the currently available heuristic search planners either do not support such variables or impose heavy restrictions on them. In particular, most admissible heuristics are restricted to domains where actions can only change numeric variables by predetermined constants. In this work, we consider the setting of optimal numeric planning with linear effects, where actions can have numeric effects that assign the result of the evaluation of a linear formula. We extend a recent formulation of Numeric LM-cut for simple effects by adding conditional effects and second-order simple effects, allowing the heuristic to produce admissible estimates for tasks with linear numeric effects. Empirical comparison shows that the proposed LM-cut heuristics favorably compete with the currently available state-of-the-art heuristics and achieve significant improvement in coverage in the domains with second-order simple effects.

Introduction

Numeric planning is a class of AI planning problems where states contain numeric variables, preconditions of actions and the goal conditions can be given in the form of inequalities of the numeric variables, and action effects modify numeric variables according to given formulas. In 2002, it was shown that numeric planning even only with constant effects is undecidable (Helmert 2002). Yet, a number of heuristic search approaches have been developed to solve numeric planning problems with both constant and linear additive effects. Hoffmann (2003) proposed a heuristic based on the interval-based relaxation, where each variable is assigned not a single value, but an interval of possible numeric assignments. More than a decade later, Aldinger, Mattmüller, and Göbelbecker (2015) showed that the interval-based relaxation can be computed in polynomial time if numeric effects do not have cyclic dependencies. Subsequently, repetition-based relaxation heuristics, which can be computed in polynomial time with cyclic dependencies, were proposed for general numeric planning (Aldinger and Nebel 2017). In parallel, Scala et al. (2016a) developed the additive interval-based relaxation (AIBR), which ensures polynomial computational time by transforming assignment effects to increase or decrease effects. However, except for the repetition-based max heuristic by Aldinger and Nebel, these methods are limited to the satisficing setting. The progress of methods for satisficing planning gave rise to the question 'can one plan optimally in presence of numeric variables in practice?'. The answer to this question was positive: recently, multiple admissible heuristics were proposed for numeric planning (Scala et al. 2020, 2017; Piacentini et al. 2018b; Kuroiwa et al. 2021). These heuristics, however, are limited to tasks with simple effects, i.e., each action increases or decreases the value of a numeric variable by a constant.

In this paper, we extend the LM-cut heuristic for numeric planning tasks with simple conditions (SCT) (Kuroiwa et al. 2021) to linear numeric planning tasks (LT), an extension that includes linear effects which increase or decrease numeric variables by linear combinations of the numeric variables. While several heuristics exist in the satisficing setting of this formalism (Hoffmann 2003; Li et al. 2018), our heuristics are the first admissible heuristics exploiting the structure of LT. Following Kuroiwa et al., we first formalize the one-variable compilation for linear numeric planning, which transforms linear numeric conditions to comparisons of a single numeric variable and a constant. Then we introduce two relaxations of the compiled task, the first-order delete-relaxation and the second-order delete-relaxation, and propose LM-cut heuristics based on the justification graphs of each of them. In particular, the first variant of the heuristic exploits conditional effects, while the second variant makes use of second-order simple effects – linear effects that are themselves dependent on only constant increase or decrease. We summarize our contributions in Figure 1. Our experimental results show that both LM-cut variants outperform the current state-of-the-art heuristics, and the second variant excels in the domains with second-order simple effects.

Background

A *linear numeric planning* task (LT) is 5-tuple $\langle \mathcal{F}, \mathcal{N}, \mathcal{A}, s_I, G \rangle$ with a set of propositions $\mathcal{F}$, a set of numeric variables $\mathcal{N}$, a set of actions $\mathcal{A}$, initial state s_I, and goal conditions G. If there are no numeric variables, i.e., $\mathcal{N} = \emptyset$, the task is called a *classical planning* task.

A state $s = \langle s_p, s_n \rangle$ has set of propositions $s_p \subseteq \mathcal{F}$

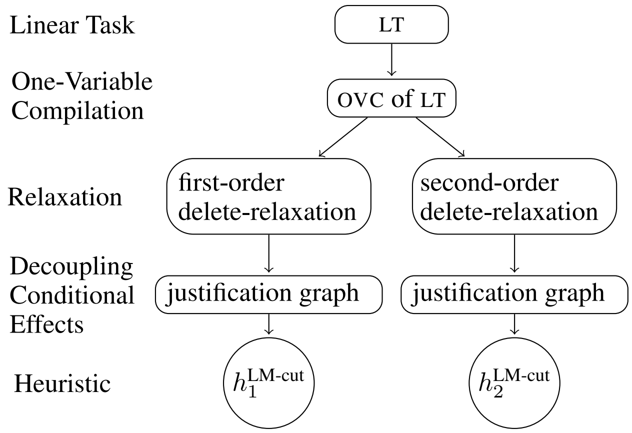

Figure 1: Overview of the steps to produce the heuristics.

and value assignments to numeric variables s_n, with $s[v]$ the value of variable v in s. A *numeric condition* ψ is a linear inequality $\sum_{v \in \mathcal{N}} w_v^\psi v \trianglerighteq w_0^\psi$, where v is a numeric variable, w_v^ψ and w_0^ψ are rational numbers, and $\trianglerighteq$ is either of $\geq$ or $>$. State s satisfies ψ, denoted by $s \models \psi$, if $\sum_{v \in \mathcal{N}} w_v^\psi s[v] \trianglerighteq w_0^\psi$. For a set of numeric conditions $\hat{\Psi}$, if $s \models \psi$ for all condition ψ in $\hat{\Psi}$, we denote $s \models \hat{\Psi}$. We say ψ is a *fact* if it is either of a proposition or a numeric condition. For a proposition ψ, we also say $s \models \psi$ if $\psi \in s_p$.

A goal condition G is pair $\langle G_p, G_n \rangle$ where G_p is a set of propositions and G_n is a set of numeric conditions. State s is a *goal state* if $G_p \subseteq s_p$ and $s \models G_n$. For a fact ψ, by abuse of notation, we say $\psi \in G$ if $\psi \in G_p$ or $\psi \in G_n$.

Action a is triplet $\langle \mathsf{pre}(a), \mathsf{eff}(a), \mathsf{cost}(a) \rangle$ where $\mathsf{cost}(a)$ is a nonnegative number. Preconditions $\mathsf{pre}(a)$ is pair $\langle \mathsf{pre}_p(a), \mathsf{pre}_n(a) \rangle$ where $\mathsf{pre}_p(a)$ is a set of propositions and $\mathsf{pre}_n(a)$ is a set of numeric conditions. Action a is applicable in state s if $\mathsf{pre}_p(a) \subseteq s_p$ and $s \models \mathsf{pre}_n(a)$, denoted by $s \models \mathsf{pre}(a)$. We use Ψ to denote the set of all numeric conditions, i.e., $\Psi = G_n \cup \bigcup_{a \in \mathcal{A}} \mathsf{pre}_n(a)$.

An effect $\mathsf{eff}(a)$ is triplet $\langle \mathsf{add}(a), \mathsf{del}(a), \mathsf{num}(a) \rangle$ where $\mathsf{add}(a)$ and $\mathsf{del}(a)$ are sets of propositions, and $\mathsf{num}(a)$ is a set of numeric effects of the form $(v \mathrel{+}= \xi + c)$, where ξ is a numeric expression of the form $\sum_{v \in \mathcal{N}} w_v^\xi v$ and c is a rational number. Each w_v^ξ is also a rational number. The value of ξ in state s is defined as $s[\xi] = \sum_{v \in \mathcal{N}} w_v^\xi s[v]$. We assume that assignment effect $v := \xi + c$ and subtractive effect $v \mathrel{-}= \xi + c$ are normalized to the additive forms $v \mathrel{+}= \xi - v + c$ and $v \mathrel{+}= -\xi - c$, and one action has at most one effect on each numeric variable. Applying action a makes state s transition to state $s[\![a]\!]$ such that $s[\![a]\!]_p = (s_p \setminus \mathsf{del}(a)) \cup \mathsf{add}(a)$, $s[\![a]\!][v] = s[v] + \xi[s] + c$ if $v \mathrel{+}= \xi + c \in \mathsf{num}(a)$, and $s[\![a]\!][v] = s[v]$ otherwise.

For state s, an *s-plan* is a sequence of actions that can be sequentially applied from s and that makes s transition to a goal state. A solution for the task is an s_I-plan, and we call it a *plan* for the task. When an s-plan is $\pi = \langle a_1, ..., a_n \rangle$, the cost of the s-plan is $\mathsf{cost}(\pi) = \sum_{i=1}^n \mathsf{cost}(a_i)$. In *optimal planning*, we find an *optimal plan*, which minimizes the cost. The value of variable v after the execution of sequence of actions π is denoted $\pi[v]$. A *heuristic* h is a function that maps state s to *heuristic value* $h(s) \in \mathbb{R}_{0+}$. If, for all state s, $h(s)$ is a lower bound of the cost of the optimal s-plan, then h is *admissible*.

Planning with Simple Numeric Conditions

Scala et al. (2016b) introduced the notion of a *simple effect* and a *simple numeric condition (*SC*)*. If an effect changes a variable only by a constant, i.e., of the form $v \mathrel{+}= c$, it is a simple effect. We call effects that are not simple in LT *linear effects*. A numeric condition $\sum_{v \in \mathcal{N}} w_v v \trianglerighteq w_0$ is a SC if for all v with $w_v \neq 0$, all effects that change v are simple effects. Tasks with SC only are called SC tasks (SCT). SCT is a subset of LT and a superset of restricted tasks (RT) (Hoffmann 2003), where every numeric condition is a comparison of a single variable and a constant, i.e., of the form $v \trianglerighteq w_0$ where v is a numeric variable and w_0 is a rational number.

SCT can be reduced to RT by introducing a new numeric variable for each SC and modifying numeric effects so that they change the variable by the net effects on the SC. Given an SCT $\Pi = \langle \mathcal{F}, \mathcal{N}, \mathcal{A}, s_I, G \rangle$, Kuroiwa et al. (2021) construct a transformed task Π^{RT} to be the 5-tuple $\langle \mathcal{F}, \mathcal{N}^{\text{RT}}, \mathcal{A}^{\text{RT}}, s_I^{\text{RT}}, G^{\text{RT}} \rangle$ defined as follows. For numeric condition $\psi : \sum_{v \in \mathcal{N}} w_v^\psi v \trianglerighteq w_0$, we add an auxiliary numeric variable $u^\psi \in \mathcal{N}^{\text{RT}}$ that corresponds to the left-hand side of the condition and replace ψ with $u^\psi \trianglerighteq w_0^\psi$. The initial value is defined as $s_I^{\text{RT}}[u^\psi] = \sum_{v \in \mathcal{N}} w_v^\psi s_I[v]$. For action $a \in \mathcal{A}$, we add a numeric effect with the form $u^\psi \mathrel{+}= \sum_{v \in \mathcal{N}} w_v^\psi c_v^a$, where $v \mathrel{+}= c_v^a$ are the original numeric effects of the action. This translation is polynomial in the number of numeric conditions of the task, and called the *one-variable compilation (*OVC*)*. For example, suppose an SCT with numeric variables x and y, condition $x + y \geq 1$, and action a with effects $x \mathrel{+}= 3$ and $y \mathrel{+}= 2$. We replace $x + y \geq 1$ with $u^{x+y} \geq 1$, and add effect $u^{x+y} \mathrel{+}= 5$ of a.

RT Π^+ is delete-free if for each action $a \in \mathcal{A}$, delete effects $\mathsf{del}(a)$ is empty and all numeric effects are of the form $v \mathrel{+}= c_v^a \in \mathsf{num}(a)$ with $c_v^a > 0$. The *support function* for such tasks is a function that maps a fact to actions that can achieve the fact and defined to be $\mathsf{supp}(\psi) = \{a \in \mathcal{A} \mid \psi \in \mathsf{add}(a)\}$ if ψ is a proposition and $\mathsf{supp}(v \trianglerighteq w_0) = \{a \in \mathcal{A} \mid v \mathrel{+}= c_v^a \in \mathsf{num}(a)\}$ if ψ is a numeric condition $v \trianglerighteq w_0$.

LM-Cut in Numeric Planning with Simple Effects

Helmert and Domshlak (2009) introduced the LM-cut heuristic for classical planning. Kuroiwa et al. (2021) adapted this heuristic to the SCT setting by compiling SCT to RT. Here, we briefly present their notations and algorithm.

The heuristic is based on disjunctive action landmarks, computed as cut in the *labelled weighted digraph* called a justification graph (JG). To construct this graph, we need to introduce additional notation. Let us start with the action multiplicator function, introduced by Scala et al. (2016) for a relaxation of $h^{\max}$ (Bonet and Geffner 2001). Intuitively, action multiplicator $\mathsf{m}_a(s, \psi)$ represents the number of times one needs to apply action a to achieve fact ψ from state s in a delete-free RT. If $s \models \psi$, $\mathsf{m}_a(s, \psi) = 0$. Otherwise, if

$a \in \mathsf{supp}(\psi)$ and ψ is a proposition, $\mathsf{m}_a(s,\psi) = 1$. If ψ is a numeric condition of the form $v \geq w$, and a has effect $v \mathrel{+}= c^a$ with $c^a > 0$, then $\mathsf{m}_a(s,\psi) = \frac{w - s[v]}{c^a}$. Condition $v > w$ is normalized to $v \geq w + \epsilon$ where ϵ is a sufficiently small constant. Otherwise, $\mathsf{m}_a(s,\psi) = \infty$.

Using this action multiplicator, Kuroiwa et al. define an inadmissible heuristic, $h^{\max}_{\mathsf{cri}}$, as a maximal fixed point of a system of recursive equations; $h^{\max}_{\mathsf{cri}}(s) = h^{\max}_{\mathsf{cri}}(s, G)$. For a set of facts F: $h^{\max}_{\mathsf{cri}}(s, F) = \max_{\psi \in F} h^{\max}_{\mathsf{cri}}(s,\psi)$. For a fact ψ, it holds that $h^{\max}_{\mathsf{cri}}(s,\psi) = 0$ if $s \models \psi$, or otherwise $h^{\max}_{\mathsf{cri}}(s,\psi) = \min_{a \in \mathsf{supp}(\psi)} h^{\max}_{\mathsf{cri}}(s, \mathsf{pre}(a)) + \mathsf{m}_a(s,\psi)\mathsf{cost}(a)$.

A *labelled weighted digraph* is formally defined by a triplet $\langle N, E, \mathsf{W}\rangle$, where N are the vertices of the graph, $E \subseteq N \times N \times A$ are labelled edges of the graph, where A denotes the label set, and $\mathsf{W} : E \to \mathbb{R}_{0+}$ is the weight function on edges. To construct the JG, each action should be associated with at most one fact in its preconditions. To this end, the precondition choice function (pcf) is defined. In the case of $h^{\max}_{\mathsf{cri}}$ this function is defined as follows:

$$\mathsf{pcf}_{\mathsf{cri}}(s,a) \in \arg\max_{\psi \in \mathsf{pre}(a)} h^{\max}_{\mathsf{cri}}(s,\psi).$$

For actions with no precondition, we add an artificial fact denoted by $\emptyset$. If $\mathsf{pre}(a) = \emptyset$, we write $\mathsf{pcf}_{\mathsf{cri}}(s,a) = \emptyset$. Thus, for each state s we can construct the following JG $\langle N, E, \mathsf{W}\rangle$, where $N = \{n_\psi \mid \psi \in \mathcal{F} \cup \Psi \cup \{\emptyset\}\}$ is the set of vertices, $E = \hat{E} \cup \{(n_\psi, n_{\psi'}; a) \mid a \in \mathsf{supp}(\psi'), \psi = \mathsf{pcf}(s,a)\}$ is the set of labeled edges with the zero-cost edges $\hat{E} = \{(n_\emptyset, n_\psi; a_0) \mid s \models \psi\}$, and the weight function is defined as $\mathsf{W}((n_\psi, n_{\psi'}; a)) = \mathsf{m}_a(s,\psi') \cdot \mathsf{cost}(a)$.

Given two disjoint vertex sets $N_1, N_2 \subseteq N$, we define a *directed cut* to be $(N_1, N_2) = \{(n_1, n_2; a) \in E \mid n_1 \in N_1, n_2 \in N_2\}$. The weights of the cut L is defined as $\mathsf{W}(L) = \min_{e \in L} \mathsf{W}(e)$. For goal condition $g \in G$ we define the vertex sets for the cut: the *goal zone* is

$$N^g = \{n_\psi \in N \mid \exists \text{ zero-weight path from } n_\psi \text{ to } n_g\};$$

the *before-goal zone* is a set of vertices that can be reached from the vertex $n_\emptyset$ without passing through N^g:

$$N^0 = \{n_\psi \in N \mid {}^{n_\psi}\ \text{is reachable from}\ {}_{n_\emptyset} \ \text{without crossing}\ {}_{N^g}\};$$

and the *beyond-goal zone* is $N^b = (N \setminus N^g) \setminus N^0$.

The LM-cut heuristic is computed in rounds. Set $h^{\text{LM-cut}}(s)$ value to zero, and iterate over:

1. Compute the $h^{\max}_{\mathsf{cri}}$ values of all relevant facts. If $\max_{g \in G} h^{\max}_{\mathsf{cri}}(g) = 0$ return the $h^{\text{LM-cut}}(s)$ value. If there is $g \in G$ such that $h^{\max}_{\mathsf{cri}}(g) = \infty$ return ∞.
2. Use the $h^{\max}_{\mathsf{cri}}$ values to construct a JG and use this graph to compute a directed cut $L = (N^0, N^g)$ with the cost $\mathsf{W}(L)$. Update $h^{\text{LM-cut}}(s) \mathrel{+}= \mathsf{W}(L)$.
3. Reduce the costs of actions so that weights of all edges in L are reduced by $\mathsf{W}(L)$.
4. Go to Step 1, using the updated action costs.

The admissibility is guaranteed by cost-partitioning (Katz and Domshlak 2008; Yang et al. 2008); the weight of the cut L, $\mathsf{W}(L)$ in 2., is admissible in a cost-partitioned task at each iteration, thus $\sum \mathsf{W}(L)$ is admissible for the original task.

One-Variable Compilation of Linear Effects

We extend the LM-cut procedure to LT. Kuroiwa et al. presented an LM-cut version that can deal with simple effects. Here we generalize it to account for non-simple effects. The LM-cut for SCT is based on OVC, the compilation of SCT into RT, so we first generalize OVC to LT as shown in Figure 1. Then, we introduce the first-order delete-relaxation, where a linear effect is relaxed to a conditional effect that increases a variable to infinity if the linear formula is positive. In the justification graphs (JGs), these conditional effects are decoupled to different actions sharing the same label. We propose heuristic $h^{\text{LM-cut}}_1$ based on such JGs. Next, we tighten the first-order delete-relaxation by preserving second-order simple effects, linear effects that only contain numeric variables affected by simple effects. In the resulting second-order delete-relaxation, we derive a lower bound to achieve a numeric condition using only simple and second-order simple effects. Using this bound, we propose heuristic $h^{\text{LM-cut}}_2$ based on augmented JGs.

Since LT is different from SCT only in that it has linear effects, in OVC of LT, we introduce additional linear effects so that they appropriately change auxiliary variables in the compiled task. An OVC of a linear numeric planning task $\Pi = \langle \mathcal{F}, \mathcal{N}, \mathcal{A}, s_I, G\rangle$ is defined as $\Pi_{\text{OVC}} = \langle \mathcal{F}, \tilde{\mathcal{N}}, \tilde{\mathcal{A}}, \tilde{s_I}, \tilde{G}\rangle$. The set of variables $\tilde{\mathcal{N}} = \mathcal{N} \cup \mathcal{N}^c$ where $\mathcal{N}^c$ is a set of auxiliary variables; for each numeric condition $\psi \in \Psi$, we add an auxiliary numeric variable $u^\psi \in \mathcal{N}^c$ with $\tilde{s_I}[u^\psi] = \sum_{v \in \mathcal{N}} w^\psi_v s_I[v]$. We introduce action $\tilde{a} \in \tilde{\mathcal{A}}$ for each $a \in \mathcal{A}$, where $\mathsf{pre}_p(\tilde{a}) = \mathsf{pre}_p(a)$, $\mathsf{pre}_n(\tilde{a}) = \{u^\psi \trianglerighteq w^\psi_0 \mid \psi \in \mathsf{pre}_n(a)\}$, $\mathsf{add}(\tilde{a}) = \mathsf{add}(a)$, $\mathsf{del}(\tilde{a}) = \mathsf{del}(a)$, and $\mathsf{cost}(\tilde{a}) = \mathsf{cost}(a)$. Goal conditions $\tilde{G}$ is pair $\langle G_P, \tilde{G}_n\rangle$, where $\tilde{G}_n = \{u^\psi \trianglerighteq w^\psi_0 \mid \psi \in G_n\}$. The set of all numeric conditions in Π_{OVC} is defined as $\tilde{\Psi} = \{u^\psi \trianglerighteq w^\psi_0 \mid \psi \in \Psi\}$.

For numeric effects of actions in $\tilde{\mathcal{A}}$, we start by dividing the effects of the original actions into simple effects and additive linear effects. Note that this is an abuse of notation and does not contradicts that an action has at most one effect on one numeric variable. We divide the effect $v \mathrel{+}= \xi + c \in \mathsf{num}(a)$ into two effects $v \mathrel{+}= c$ and $v \mathrel{+}= \xi$. The set of simple effects of an action $a \in \mathcal{A}$ is defined as $\mathsf{num}_1(a) = \{v \mathrel{+}= c \in \mathsf{num}(a) \mid c \in \mathbb{Q}\}$. The non-simple effects of an action a are denoted $\mathsf{num}^c_1(a) = \mathsf{num}(a) \setminus \mathsf{num}_1(a)$. The set of all actions that have only simple effects is defined as

$$\mathcal{A}_1 = \{a \in \mathcal{A} \mid \mathsf{num}^c_1(a) = \emptyset\}, \text{ and } \mathcal{A}^c_1 := \mathcal{A} \setminus \mathcal{A}_1.$$

In each action $\tilde{a} \in \tilde{\mathcal{A}}$, the original effects stay the same: each effect of the form $v \mathrel{+}= \xi + c \in \mathsf{num}(a)$ is divided into $v \mathrel{+}= c \in \mathsf{num}_1(\tilde{a})$ and $v \mathrel{+}= \xi \in \mathsf{num}^c_1(\tilde{a})$. For each auxiliary variable $u^\varphi \in \mathcal{N}^c$ that corresponds to $u^\varphi = \sum_{v \in \mathcal{N}} w^\varphi_v v$, we add the following constant and linear additive effects

$$u^\varphi \mathrel{+}= \sum_{v \mathrel{+}= c_v \in \mathsf{num}_1(\tilde{a})} w^\varphi_v c_v = c^a_{u^\varphi} \in \mathsf{num}_1(\tilde{a})$$

$$u^\varphi \mathrel{+}= \sum_{v \mathrel{+}= \xi_v \in \mathsf{num}^c_1(\tilde{a})} w^\varphi_v \sum_{v' \in \mathcal{N}} w^{\xi_v}_{v'} v' \in \mathsf{num}^c_1(\tilde{a}).$$

The expression in the second line is a linear formula in the original variables in $\mathcal{N}$, and we denote it by $\phi^a_{u^\xi}$. Thus, we can write $u^\xi \mathrel{+}= \phi^a_{u^\xi} + c^a_{u^\xi} \in \mathsf{num}(\tilde{a})$.[1]

Example 1. *Let $\Pi = \langle \mathcal{F}, \mathcal{N}, \mathcal{A}, s_I, G\rangle$ be a linear numeric planning task, where $\mathcal{F} = \emptyset$, $\mathcal{N} = \{x, y\}$, $\mathcal{A} = \{a_1, a_2\}$, $s_I = \{x = 1, y = 0\}$, and $G = \{2y \geq 30\}$. For all actions $a \in \mathcal{A}$, $\mathsf{pre}_p(a) = \mathsf{pre}_n(a) = \mathsf{add}(a) = \mathsf{del}(a) = \emptyset$, $\mathsf{cost}(a) = 1$, and $\mathsf{num}(a)$ is defined as:*

$\mathcal{A}$	num	*type*
a_1	$\{x \mathrel{+}= 1\}$	*simple (constant increase/decrease)*
a_2	$\{y \mathrel{+}= 3x\}$	*linear (linear formula)*

In Π_{OVC}, auxiliary variables are introduced for numeric conditions. In this example, the variable u^{2y} is introduced for $2y \geq 30$, so $\mathcal{N}^c = \{u^{2y}\}$. $\tilde{s_I}[u^{2y}] = 0$ and $\tilde{G}_n = \{u^{2y} \geq 30\}$. The numeric effects are

$\tilde{\mathcal{A}}$	num
$\tilde{a}_1$	$\{x \mathrel{+}= 1\}$
$\tilde{a}_2$	$\{y \mathrel{+}= 3x, u^{2y} \mathrel{+}= 6x\}$

For u^{2y}, since a_2 has linear effect $y \mathrel{+}= 3x$, linear effect $u^{2y} \mathrel{+}= 2{\cdot}3x$ is introduced, which is represented by original variable $x \in \mathcal{N}$.

Relaxing Linear Effects

Kuroiwa et al. presented the delete-relaxation of RT that relaxed simple numeric effects by removing negative constant effects. For each action $a \in \mathcal{A}$, they remove all simple numeric effects of the form $u \mathrel{+}= c$ where $u \in \tilde{\mathcal{N}}$ and $c \leq 0$. Propositional delete-effects are removed as well.

We extend the delete-relaxation of RT to OVC of LT. In OVC, since a numeric condition has the form $u \geq w$ where u is a numeric variable and w is a constant, ignoring negative numeric effects on u underestimates the effort to achieve the condition, consistent with admissibility. However, in LT, the values of linear effects are state dependent, and we do not know whether an effect is negative in advance. To address this issue, we ignore negative effects by turning linear effect $v \mathrel{+}= \xi$ into a conditional effect which fires only if $\xi > 0$. In addition, to account for the effort of achieving a numeric condition $v \geq w$ via $v \mathrel{+}= \xi$, we overestimate the effect $v \mathrel{+}= \xi$ as $v \mathrel{+}= \infty$. In other words, all conditions of the form $v \geq w$ are achieved by applying the effect $v \mathrel{+}= \infty$ exactly once, again underestimating the effort.

Formally, conditional effects of an action a are given as a set of tuples, where each conditional effect e is represented as $\langle \mathsf{cond}(e), \mathsf{eff}(e)\rangle$, and $\mathsf{cond}(e)$ and $\mathsf{eff}(e)$ have the structure of preconditions and an effect of a regular action. The conditional effect is applied in a state s immediately after the application of the action, but only if $s \models \mathsf{cond}(e)$.[2]

We introduce the *first-order delete-relaxation* of Π_{OVC}, $\Pi^1_{\text{OVC}} = \langle \mathcal{F}, \mathcal{N}^1, \mathcal{A}^1, s^1_I, \tilde{G}\rangle$ where $\mathcal{N}^1 = \mathcal{N} \cup \mathcal{N}^{c,1}$. In $\mathcal{N}^{c,1}$, in addition to the auxiliary variables in $\mathcal{N}^c$, we introduce $u^\xi, u^{-\xi} \in \mathcal{N}^{c,1}$ for each linear effect $v \mathrel{+}= \xi \in \mathsf{num}^c_1(a)$ for each $a \in \mathcal{A}$. Here, $v \in \mathcal{N}$, and we do not introduce auxiliary variables for linear effects on auxiliary variables in $\mathcal{N}^c$. The domain of each variable $v \in \mathcal{N}^1$ is extended to $\mathbb{R} \cup \{\infty\}$, where for each $c \in \mathbb{R}$ it holds that $\infty + c = \infty$, $\infty \cdot c = \infty$, and $c \leq \infty$. It holds that $\infty + \infty = \infty$ and $\infty \cdot \infty = \infty$. The initial values of the auxiliary variables are defined as $s^1_I[u^\xi] = \sum_{v \in \mathcal{N}} w^\xi_v s_I[v]$ and $s^1_I[u^{-\xi}] = -s_I[u^\xi]$. We have action $a^1 \in \mathcal{A}^1$ for each $a \in \mathcal{A}$ with $\mathsf{pre}(a^1) = \mathsf{pre}(\tilde{a})$, $\mathsf{add}(a^1) = \mathsf{add}(a)$, $\mathsf{del}(a^1) = \emptyset$, and $\mathsf{cost}(a^1) = \mathsf{cost}(a)$. The simple effects on the original and auxiliary variables are defined as the same as OVC, but negative effects are removed. For an action $a \in \mathcal{A}^c_1$, we replace each linear effect $v \mathrel{+}= \xi \in \mathsf{num}^c_1(a)$ with conditional effects e^{v+}_a and e^{v-}_a

$$\begin{aligned}\mathsf{cond}(e^{v+}_a) &= \langle \emptyset, \{u^\xi > 0\}\rangle,\\ \mathsf{eff}(e^{v+}_a) &= \langle \emptyset, \emptyset, \{v \mathrel{+}= \infty\} \cup \bigcup_{u^\varphi \in \mathcal{N}^{c,1} : w^\varphi_v > 0} \{u^\varphi \mathrel{+}= \infty\}\rangle,\end{aligned}$$

$$\begin{aligned}\mathsf{cond}(e^{v-}_a) &= \langle \emptyset, \{u^{-\xi} > 0\}\rangle,\\ \mathsf{eff}(e^{v-}_a) &= \langle \emptyset, \emptyset, \bigcup_{u^\varphi \in \mathcal{N}^{c,1} : w^\varphi_v < 0} \{u^\varphi \mathrel{+}= \infty\}\rangle.\end{aligned}$$

The set of all numeric conditions is extended to $\Psi^1 = \tilde{\Psi} \cup \{u^\xi > 0, u^{-\xi} > 0 \mid v \mathrel{+}= \xi \in \mathsf{num}^c_1(a), a \in \mathcal{A}^c_1\}$. The asymmetry of e^{v+}_a and e^{v-}_a comes from the fact that $\varphi = -v$ is a linear formula. Thus, in practice, v and $u^\varphi = -v$ are treated in the same way.

Our relaxation is similar to AIBR (Scala, Haslum, and Thiébaux 2016) in that assignment effects are normalized to additive effects and linear effects are divided into two conditional infinite additive effects. Alternatively, $\mathcal{A}^1$ can be seen as a special case of the effect-abstraction based relaxation (Li et al. 2018), where linear effects are abstracted by intervals $(-\infty, 0)$ and $(0, \infty)$.

Example 2. *Using the task Π from Ex. 1, $\mathcal{A}^1$ is as follows:*

$\mathcal{A}^1$	cond	eff
a^1_1	$\emptyset$	$\{x \mathrel{+}= 1, u^{3x} \mathrel{+}= 3\}$
a^1_2	$\{u^{3x} > 0\}$	$\{y \mathrel{+}= \infty, u^{2y} \mathrel{+}= \infty\}$

Auxiliary variables u^{3x} and u^{-3x} are introduced, but u^{6x} is not introduced since $u^{2y} \mathrel{+}= 6x$ is a linear effect on an auxiliary variable. With condition $u^{3x} > 0$, in addition to $y \mathrel{+}= \infty$, effect $u^{2y} \mathrel{+}= \infty$ is introduced as linear expression $2y$ has a positive coefficient of y. In contrast, since no linear expression has a negative coefficient on y, there is no effect with condition $u^{-3x} > 0$. Note that $u^{-3x} \mathrel{+}= -3$ is removed from $\mathsf{num}(a^1_1)$ since it decreases the value of u^{-3x}.

From the Relaxation to LM-Cut Heuristic

Since the first-order delete-relaxation employs conditional effects, the classical LM-cut procedure cannot be directly applied. Keyder, Hoffmann, and Haslum (2012) proposed a decoupling method for LM-cut to handle conditional effects in classical planning. Since this LM-cut heuristic does not dominate $h^{\max}$ in theory, Röger, Pommerening, and Helmert (2014) proposed another LM-cut heuristic handling

[1] In what follows we omit the indices if evident from the context.

[2] The usual assumption in the case of conditional effects is that their application is consistent and not order dependent.

conditional effects, which dominates $h^{\max}$. However, the former empirically performs better than the latter (Röger, Pommerening, and Helmert 2014). Therefore, we employ the method proposed by Keyder, Hoffmann, and Haslum.

Let a^1 be an action in $\mathcal{A}^1$ of the form

$$a^1 = \langle \mathsf{pre}(a^1), \langle \mathsf{add}(a^1), \emptyset, \mathsf{num}_1(a^1), \mathsf{condeff}(a^1)\rangle\rangle.$$

We decouple this action into a set of actions with regular effects. The actions with the conditional effects removed, i.e., $\langle \mathsf{pre}(a), \langle \mathsf{add}(a), \emptyset, \mathsf{num}_1(a^1)\rangle$, are named *core actions* and are denoted by $\mathcal{A}^1_{\text{core}}$. All conditional effects are transformed into regular actions, i.e.,

$$\forall \langle \mathsf{cond}(e_a), \mathsf{eff}(e_a)\rangle \in \mathsf{condeff}(a^1) :$$
$$a^{e,\infty} = \langle \mathsf{pre}(a) \cup \mathsf{cond}(e_a), \mathsf{eff}(e_a)\rangle.$$

The set of these non-core actions is denoted by $\mathcal{A}^1_{\text{cond}}$. Note that all effects of action $a^{e,\infty}$ are of the form $u \mathrel{+}= \infty$ where $u \in \mathcal{N}^1$. The set of all actions of the transformed task is $\mathcal{A}^1_{\text{JG}} = \mathcal{A}^1_{\text{core}} \cup \mathcal{A}^1_{\text{cond}}$. All actions that originate from the action a share the same label in terms of cost, i.e., reducing the cost of a will change the cost of all derivative actions.

Next, we extend the action multiplicator to account for the 'plus infinity' numeric effects: if an action a has the effect $v \mathrel{+}= \infty$, we define $\mathsf{m}_a(s, \psi) = 1$ for each numeric condition $\psi : v \geq w$.

The LM-cut heuristic, $h_1^{\text{LM-cut}}$, is the same as in SCTs but the JG is constructed based on $\mathcal{A}^1_{\text{JG}}$, where core-actions and their conditional effects share the same label and action cost. As in the original LM-cut, at each iteration, a non-zero weight cut L is extracted from the JG, the heuristic value is increased by $\mathsf{W}(L)$, and the cost of action a is reduced by $\mathsf{cost}_1(a)$. This procedure ensures cost-partitioning. The cost function cost_1 is defined so that the weights of edges in L are reduced by $\mathsf{W}(L)$.

To show that LM-cut is admissible, we prove that the weight of a cut in the JG is an admissible estimate of the cost-partitioned task.

Theorem 1. *Let Π_{OVC} be the* OVC *of a solvable* LT *with a non-zero optimal cost. Let L be a directed cut in a JG of the first-order delete-relaxation Π^1_{OVC}, where the set of actions in the cut is given by $\mathsf{lbl}(L) = \{a \mid (n_1, n_2; a) \in L\}$. For action a, let the minimum of multiplicators in the cut be $\mathsf{m}^L_a = \min_{(n_\psi, n_{\psi'}; a) \in L} \mathsf{m}_a(s, \psi')$ and cost_1 is defined as*

$$\mathsf{cost}_1(a) = \begin{cases} \frac{\mathsf{W}(L)}{\mathsf{m}^L_a} & \textit{if } a \in \mathsf{lbl}(L) \\ 0 & \textit{otherwise.} \end{cases}$$

Let $\Pi^1_{\text{OVC},1}$ be a copy of Π^1_{OVC} except that action a has cost $\mathsf{cost}_1(a)$. Then, the weight of the cut $\mathsf{W}(L) = \min_{e \in L} \mathsf{W}(e)$ is admissible for $\Pi^1_{\text{OVC},1}$.

Proof Sketch: The proof is an extension of the proof of Thm.1 in Kuroiwa et al. (2021) to account for the 'plus infinity' effects. This case resembles the case of regular LM-cut, from the perspective of the 'plus infinity' action, the fact $v = \infty$ is binary, and is either achieved or not. See the supplementary material (SM) (Kuroiwa, Shleyfman, and Beck 2022) for details. □

Exploiting Second-Order Simple Effects

In the first-order delete-relaxation, all linear effects are replaced with 'plus infinity' effects. However, it is possible to obtain a tighter relaxation for particular types of linear effects, which we call second-order simple effects.

Motivating Example

In the task Π in Ex. 1, to achieve the goal condition $2y \geq 30$, we need to apply a_2 to increase $2y$. In the first-order delete-relaxation Π^1_{OVC} in Ex. 2, we compile the goal condition to $u^{2y} \geq 30$ and relax the effect of a_2 to $u^{2y} \mathrel{+}= \infty$ with condition $u^{3x} > 0$. Since $s_I[x] = 1$, $s^1_I[u^{3x}] = 3 > 0$, plan $\langle a^1_2 \rangle$ achieves the goal condition, and the optimal cost is estimated by $\mathsf{cost}(a^1_2) = 1$.

In the following example, we introduce an approach that provides a better estimation than the one given by Π^1_{OVC}. The idea is similar to the operator-counting approach (Pommerening et al. 2014). Since a_1 increases x, a better approximation first applies a_1 several times and only then apply a_2. Let X_{a_1} and X_{a_2} be the number of applications of a_1 and a_2. We minimize the cost $X_{a_1}\mathsf{cost}(a_1) + X_{a_2}\mathsf{cost}(a_2)$ of achieving $2y \geq 30$. Since x is increased by a constant, we represent the value of $2y$ as $0 + 2 \cdot X_{a_2} \cdot 3(1 + X_{a_1}) = 6X_{a_2}(1 + X_{a_1})$. The optimal cost to achieve the goal condition, $2y \geq 30$, is the optimal solution of the following optimization problem:

$$\min \; X_{a_1}\mathsf{cost}(a_1) + X_{a_2}\mathsf{cost}(a_2) \tag{1}$$
$$\text{s.t. } 6X_{a_2}(1 + X_{a_1}) \geq 30 \tag{2}$$
$$X_{a_1}, X_{a_2} \in \mathbb{Z}_{0+}. \tag{3}$$

This problem is a nonlinear integer programming problem, which is computationally expensive to solve in general. Instead of directly solving the problem, we represent a lower bound of the optimal cost of the problem as a closed-form formula. First, we relax the integrality of variables X_{a_1} and X_{a_2}. Then, we decompose the problem into two phases: first, increase x using a_1 to constant $C > 0$; then, increase y using a_2. In the second step, y is increased by C each time a_2 is applied, and Constraint (2) is replaced with $6X_{a_2} \cdot C \geq 30$. To minimize the objective, we need to minimize X_{a_2}, so $X_{a_2} = \frac{5}{C}$ is the optimal solution. The problem is reformulated as follows:

$$\min \; X_{a_1}\mathsf{cost}(a_1) + \frac{5}{C}\mathsf{cost}(a_2) \tag{4}$$
$$\text{s.t. } 1 + X_{a_1} = C \tag{5}$$
$$X_{a_1} \geq 0, C > 0. \tag{6}$$

Since $C = 1 + X_{a_1}$ and $\mathsf{cost}(a_1) = \mathsf{cost}(a_2) = 1$, the objective is represented as $X_{a_1}\mathsf{cost}(a_1) + \frac{5}{1+X_{a_1}}\mathsf{cost}(a_2) = X_{a_1} + \frac{5}{1+X_{a_1}}$. Here the objective is a function of single variable X_{a_1}, so we can compute the minimum using a basic method in calculus. By differentiating the objective, we get the derivative $1 - \frac{5}{(1+X_{a_1})^2}$. The objective takes an extreme value when the derivative is zero, i.e., $X_{a_1} = \sqrt{5} - 1$ and $X_{a_1} = -\sqrt{5} - 1$. When $X_{a_1} > 0$, the derivative is positive if $X_{a_1} > \sqrt{5} - 1$ and negative if $X_{a_2} < \sqrt{5} - 1$. Therefore, the minimum value is achieved when $X_{a_1} = \sqrt{5} - 1$.

Substituting the equations, $C = \sqrt{5}$ and $X_{a_2} = \sqrt{5}$. Thus, the lower bound on the cost to achieve the goal condition is $2\sqrt{5} - 1$, which is better than the one estimated in Π^1_{OVC}.

Second-Order Simple Effects

In the example, we compute a lower bound to achieve a numeric condition using a linear effect in a closed-form. The solution of the continuous relaxation of the non-linear optimization problem (1)–(3) is possible only because the linear effect can be changed at most by a constant, resulting in a single quadratic inequality (2). Focusing on this point, we define simple variables: as those that are changed only by constants.

Definition 1 (Simple Variable)**.** *We say that a variable v is simple if it can be changed by only simple effects. The set of all simple variables of the task Π is denoted by $\mathcal{N}_1$.*

We define a type of linear effect that use only simple variables.

Definition 2 (Second-Order Simple Effects (SOSE))**.** *The effect $v \mathrel{+}= \xi \in \mathsf{num}^c_1(a)$ of action a is a SOSE if*

1. all variables in ξ are simple variables; and
2. all actions that change variables in ξ do not change v.

The set of all SOSE of action a is denoted by $\mathsf{num}_2(a)$.

In Ex. 1, variable x is simple since x is only changed by action a_1 with effect $x \mathrel{+}= 1$. Variable y is not simple as a_2 has effect $y \mathrel{+}= 3x$, which is a SOSE since x is a simple variable and a_1 does not have effects on y.

The second condition in Def. 2 is necessary for computing the bound as in the motivating example; in the example, we build Constraint (2) based on the idea that we should apply a_1 first to increase x and then apply a_2 to increase y to achieve condition $2y \geq 30$. Now, suppose that a_1 had effect $y \mathrel{+}= z$ and a_2 had effect $z \mathrel{+}= 3$. We could no longer determine which of a_1 or a_2 to apply first. Thus, we exclude such a case by imposing the second condition to a SOSE.

For action a, we divide its numeric effects into three sets $\mathsf{num}(a) = \mathsf{num}_1(a) \cup \mathsf{num}_2(a) \cup \mathsf{num}^c_{1,2}(a)$, where $\mathsf{num}_1(a)$ is the set of simple effects of the form $v \mathrel{+}= c$, $\mathsf{num}_2(a)$ is the set of SOSE of the form $v \mathrel{+}= \xi$, and $\mathsf{num}^c_{1,2}(a) = \mathsf{num}(a) \setminus (\mathsf{num}_1(a) \cup \mathsf{num}_2(a))$ are all other effects.

A Relaxation Preserving SOSE

In the first-order delete-relaxation, we relaxed all linear effects by conditionally adding an infinity to the variable affected by a positive linear formula. Now, we differentiate between SOSE and non-SOSE linear effects. We define the *second-order delete-relaxation* of Π_{OVC}, $\Pi^2_{\text{OVC}} = \langle \mathcal{F}, \mathcal{N}^2, \mathcal{A}^2, s^2_I, \tilde{G} \rangle$ where $\mathcal{N}^2 = \mathcal{N} \cup \mathcal{N}^{c,2}$. In $\mathcal{N}^{c,2}$, we introduce auxiliary variables representing SOSE in addition to variables in $\mathcal{N}^{c,1}$. Note that for a SOSE on an original variable, i.e., $v \mathrel{+}= \xi$ where $v \in \mathcal{N}$, the corresponding auxiliary variables are already introduced in $\mathcal{N}^{c,1}$. Therefore, we introduce $u^\xi, u^{-\xi} \in \mathcal{N}^{c,2}$ for SOSE $u \mathrel{+}= \xi$ where $u \in \mathcal{N}^{c,1}$ is an auxiliary variable. Since variables in SOSE ξ are changed by only simple effects, u^ξ is a simple variable and changed by only simple effects. Thus, we do not add auxiliary variables $u^\phi, u^{-\phi}$ for linear effect $u^\xi \mathrel{+}= \phi$ since such an effect does not exist.

For each $a \in \mathcal{A}$, we introduce $a^2 \in \mathcal{A}^2$, where $\mathsf{pre}(a^2) = \mathsf{pre}(\tilde{a})$, in which original numeric conditions are replaced with the OVC versions, $\mathsf{add}(a^2) = \mathsf{add}(a)$, $\mathsf{del}(a^2) = \emptyset$, and $\mathsf{cost}(a^2) = \mathsf{cost}(a)$. Similarly to $\mathcal{A}^1$, simple effects on auxiliary variables are defined as the same as OVC, but negative effects are removed. We divide the linear effects into three cases: SOSE, non-SOSE, and auxiliary variables.

1. Let $v \mathrel{+}= \xi \in \mathsf{num}_2(\tilde{a})$ be a SOSE. To assure that the value of v can only grow we replace this effect by $v \mathrel{+}= u^\xi$ with condition $u^\xi > 0$. We also have $v \mathrel{+}= u^{-\xi}$ with condition $u^{-\xi} > 0$, which is also a SOSE. The set of all numeric conditions is now defined as $\Psi^2 = \Psi^1 \cup \{u^\xi > 0, u^{-\xi} > 0 \mid u \mathrel{+}= \xi \in \mathsf{num}_2(\tilde{a}), u \in \mathcal{N}^{c,1}, \tilde{a} \in \tilde{\mathcal{A}}\}$.
2. In the non-SOSE case, $v \mathrel{+}= \xi \in \mathsf{num}^c_{1,2}(a)$, where v is an original variable, as in $\mathcal{A}^1$, we add the conditional effect $v \mathrel{+}= \infty$, in the case when $u^\xi > 0$.
3. By construction of OVC, each auxiliary variable $u^\varphi = \sum_{v \in \mathcal{N}} w^\varphi_v v$ is affected by action a only if there is an original variable $v \in \mathcal{N}$ that is affected by action a and $w^\varphi_v \neq 0$. Let $v \mathrel{+}= \xi_v \in \mathsf{num}^c_1(a)$ be such a linear effect. For each auxiliary variable u^φ the effect
$$u^\varphi \mathrel{+}= \sum_{v \mathrel{+}= \xi_v \in \mathsf{num}^c_1(a)} w^\varphi_v \xi_v \in \mathsf{num}^c_{1,2}(\tilde{a}),$$
is replaced by the effect $u^\varphi \mathrel{+}= \infty$ with condition $u^{\xi_v} > 0$, if $w^\varphi_v > 0$. In the case when $w^\varphi_v < 0$ the same effect is replaced by $u^\varphi \mathrel{+}= \infty$ with condition $u^{-\xi_v} > 0$.

Example 3. *Using the task Π from Ex. 1, $\mathcal{A}^2$ is as follows:*

$\mathcal{A}^2$	cond	eff
a^2_1	$\emptyset$	$\{x \mathrel{+}= 1, u^{3x} \mathrel{+}= 3, u^{6x} \mathrel{+}= 6\}$
a^2_2	$\{u^{3x} > 0\}$	$\{y \mathrel{+}= u^{3x}\}$
	$\{u^{6x} > 0\}$	$\{u^{2y} \mathrel{+}= u^{6x}\}$

Since x is a simple variable, effects $y \mathrel{+}= 3x$ and $u^{2y} \mathrel{+}= 6x$ are SOSE and converted to $y \mathrel{+}= u^{3x}$ and $u^{2y} \mathrel{+}= u^{6x}$, where u^{6x} is an additional auxiliary variable. The set of auxiliary variables is $\mathcal{N}^{c,2} = \{u^{2y}, u^{3x}, u^{-3x}, u^{6x}, u^{-6x}\}$.

A Lower Bound on SOSE

Our goal now is to estimate from below the optimal cost of achieving a numeric fact using first- and second-order effects in a second-order delete-relaxation. To this end, in Thm. 2 we introduce a set of local minima and show that one of these local minima constitutes a global minimum that can be used as a lower bound estimate.

We start with introducing an inequality that describes the upper bound on the value that a numeric variable can achieve given a predetermined set of actions. This inequality corresponds to Constraint (2) in the example. Note also that this inequality reasons about the best possible order of application of a given set of actions while ignoring the preconditions of these actions.

Lemma 1. *Let Π^2_{OVC} be the second-order relaxation of an* LT*, with the set of actions $\mathcal{A}^2$. Suppose that numeric condition $v \geq w_0$ is achieved by sequence of actions π from state* *s, and v is changed by only simple effects and SOSE. By X_a we denote the number of times action a appears in π. Then,*

$$w_0 \le s[v] + \sum_{\substack{a \in \pi : v += c_v^a \\ \in \mathsf{num}_1(a)}} c_v^a X_a + \sum_{\substack{a \in \pi : v += u \\ \in \mathsf{num}_2(a)}} X_a \left(s[u] + \sum_{\substack{\hat{a} \in \pi : u += c_u^{\hat{a}} \\ \in \mathsf{num}_1(\hat{a})}} c_u^{\hat{a}} X_{\hat{a}} \right). \tag{7}$$

Proof Sketch: We obtain (7) by concentrating only on action effects that either affect the variable v directly or via SOSE. We ignore preconditions, and recall that for an effect $v \mathrel{+}= u$ to be SOSE all actions that change u cannot change v. We reorder actions with the simple effects and SOSE that affect the variable v in a manner that maximizes its resulting value. The full proof by induction is provided in the SM.

The intuition for the claim is as follows: since v is affected only by simple effects or SOSE, an order of application that yields the maximal value for v is the following: increase the simple variables in $\mathcal{N}_1^2$ that affect v via SOSE, and then apply the actions that affect v directly. □

We now prove the main theoretical result of this paper: a lower bound on the cost of achieving a numeric condition via simple effects and SOSE. Intuitively, we obtain the lower bound by solving an optimization problem similar to (1)–(3). However, since the optimization problem we deal with is not linear (cf. Constraint (7)), we compute three sets of local minima on the cost of achieving a numeric fact and then theoretically show that the global minimum must lie within one of these sets. The size of these sets is at most quadratic in the number of actions involved. In the following subsection, we show how to exploit this global minimum in the LM-cut heuristic.

Theorem 2. *Let Π^2_{OVC} be the second-order relaxation of an* LT*, with the set of actions $\mathcal{A}^2$. Suppose that numeric condition $v \ge w_0$ is achieved from state s, and v is changed by only simple effects and SOSE. The cost to achieve $v \ge w_0$ is bounded from below by* $\inf M_1 \cup M_2 \cup M_3$*, where*

$$\begin{aligned}
M_1 &= \{\frac{w_0 - s[v]}{c}\mathsf{cost}(a) \mid v \mathrel{+}= c \in \mathsf{num}_1(a), a \in \mathcal{A}^2\}, \\
M_2 &= \{\frac{w_0 - s[v]}{c + s[u]}\mathsf{cost}(a) \mid \\
&\quad v \mathrel{+}= u + c \in \mathsf{num}(a), s[u] > 0, a \in \mathcal{A}^2\}, \\
M_3 &= \{\mathsf{m}^u_{\hat{a}_u,a}(s, v \ge w_0)\mathsf{cost}(\hat{a}_u) + \\
&\quad \mathsf{m}^v_{\hat{a}_u,a}(s, v \ge w_0)\mathsf{cost}(a) \mid \\
&\quad v \mathrel{+}= u \in \mathsf{num}_2(a), u \mathrel{+}= c \in \mathsf{num}_1(\hat{a}_u), \\
&\quad \mathsf{m}^u_{\hat{a}_u,a}(s, v \ge w_0) > 0, a \in \mathcal{A}^2\}.
\end{aligned}$$

We specify the constants $\mathsf{m}^u_{\hat{a}_u,a}(s, v \ge w_0)$ and $\mathsf{m}^v_{\hat{a}_u,a}(s, v \ge w_0)$ in the next subsection.

Proof Sketch: To obtain the required lower bound we use Lem. 1 to formulate the following optimization problem:

$$\min_{X_a \ge 0 : a \in \mathcal{A}^2} f = \sum_{a \in \mathcal{A}^2} X_a \mathsf{cost}(a),$$

under Constraint (7) in Lem. 1 changed to equality assuming that sequence of action π achieves $v \ge w_0$. Note that Constraint (7) is the only constraint that is not of the form $X_a \ge 0$, and the change is valid, since a linear function achieves its extrema on the boundaries of this set. This objective function corresponds to (1) in the example.

Since $\mathsf{cost}(a) \ge 0$ for each $a \in \mathcal{A}^2$ we can set $X_a = 0$ for each a that does not appear in the constraint. To solve this optimization we use Lagrange multipliers. Unfortunately, the direct application would require us a large number of cases. Thus, to ease the proof we divide it into two subproblems. First, using Lagrange multipliers, we evaluate the minimal cost of obtaining the value C_u for a simple variable u

$$C_u = s[u] + \sum_{\substack{\hat{a} \in \pi : u += c_u^{\hat{a}} \\ \in \mathsf{num}_1(\hat{a})}} c_u^{\hat{a}} X_{\hat{a}}$$

where $C_u > s[u]$. Intuitively, since u is a simple variable, the minimal cost to achieve C_u is obtained by an action that minimizes the ratio $\frac{\mathsf{cost}(\hat{a}_u)}{c_u^{\hat{a}_u}}$, and constitutes

$$\frac{C_u - s[u]}{c_u^{\hat{a}_u}}\mathsf{cost}(\hat{a}_u) = \mathsf{m}_{\hat{a}_u}(s, u \ge C_u)\mathsf{cost}(\hat{a}_u).$$

Then, we can declare C_u to be a variable that substitutes the expression $s[u] + \sum_{\substack{\hat{a} \in \pi : u += c_u^{\hat{a}} \\ \in \mathsf{num}_1(\hat{a})}} c_u^{\hat{a}} X_{\hat{a}}$, and once again use Lagrange multipliers to compute the cost of the following minimization problem

$$\min_{X_a \ge 0, C_u \ge s[u]} X_a \mathsf{cost}(a) + \frac{C_u - s[u]}{c_u^{\hat{a}_u}}\mathsf{cost}(\hat{a}_u),$$
$$\text{s.t. } w_0 - s[v] = X_a(c_a + C_u),$$

for each action a with SOSE $v \mathrel{+}= u + c_a \in \mathsf{num}(a)$. The solutions to these problems are captured by the set M_3, and constitute local minima for pairs of simple and SOSE actions that achieve $\psi : v \ge w_0$. To obtain the global optimum we minimize over M_3 together with M_1 and M_2, the potential minima on the cost of achieving ψ using only simple effects and only SOSE (without anything affecting the simple variables), respectively. □

LM-Cut for SOSE

Since the actions in $\mathcal{A}^2$ also employ conditional effects, we once again use the decomposition method proposed by Keyder, Hoffmann, and Haslum. We introduce $\mathcal{A}^2_{\mathsf{JG}}$. As with $\mathcal{A}^1_{\mathsf{JG}}$, we assume that decomposed conditional effects and the core-actions share label and cost in $\mathcal{A}^2_{\mathsf{JG}}$.

Let $\psi : v \ge w_0$ be a numeric condition. Let $a \in \mathsf{supp}(\psi)$ be an action that have SOSE $v \mathrel{+}= u \in \mathsf{num}_2(a)$ and simple effect $v \mathrel{+}= c_v \in \mathsf{num}_1(a)$ where $c_v \ge 0$. Let the set $\mathcal{A}^2_u$ be the set of all actions that have a simple effects on u, i.e., $\hat{a}_u \in \mathcal{A}^2_u$ iff there is $u \mathrel{+}= c_u \in \mathsf{num}_1(\hat{a}_u)$ such that $c_u > 0$. Pairs of such actions are denoted $\langle \hat{a}_u, a \rangle$, and the second-order supporters of a numeric condition ψ are given by

$$\mathsf{supp}_2(\psi) = \{\langle \hat{a}_u, a \rangle \mid v \mathrel{+}= u \in \mathsf{num}_2(a), \hat{a}_u \in \mathcal{A}^2_u\}.$$

With Thm. 2, we can now extend the definition of $\mathsf{m}_a(s,\psi)$ for $\mathcal{A}^2_{\text{JG}}$, where ψ is a fact. As in the first-order delete-relaxation, intuitively, $\mathsf{m}_a(s,\psi)$ is the number of applications of action a to achieve ψ from state s. Similarly, to achieve numeric condition $\psi : v \geq w_0$, action $\hat{a}_u$ with simple effect $u \mathrel{+}= c_u$ is first applied $\mathsf{m}^u_{\hat{a}_u,a}(s,\psi)$ times to increase the value of simple variable u, and then action a with SOSE $v \mathrel{+}= u$ is applied $\mathsf{m}^v_{\hat{a}_u,a}(s,\psi)$ times.

Definition 3 (Second-Order Action Multiplicator)**.** *Given state s, fact ψ and an action a, the multiplicator $\mathsf{m}_a(s,\psi)$ stays the same for propositional and simple numeric effects. Thus, assume that condition $\psi : v \geq w_0$ can be achieved by the SOSE $v \mathrel{+}= u \in \mathsf{num}_2(a)$, where the full effect of a on v is $v \mathrel{+}= u + c \in \mathsf{num}(a)$. We can extend the definition of action multiplicator by*

$$\mathsf{m}_a(s,\psi) = \begin{cases} \frac{w_0 - s[v]}{c} & s[u] \leq 0, c > 0 \\ \frac{w_0 - s[v]}{c + s[u]} & s[u] > 0, c \geq 0 \end{cases}$$

and $\mathsf{supp}(v \geq w_0) = \{a \in \mathcal{A}^2 \mid v \mathrel{+}= c \in \mathsf{num}_1(a)\} \cup \{a \in \mathcal{A}^2 \mid v \mathrel{+}= u + c \in \mathsf{num}_2(a), s[u] > 0\}$. *To obtain the lower bound on achieving $v \geq w_0$, we also need to evaluate the action pairs $\langle \hat{a}_u, a\rangle$, where a has the SOSE $v \mathrel{+}= u \in \mathsf{num}_2(a)$, and $\hat{a}_u \in \mathcal{A}^2_u$ has the simple effect $u \mathrel{+}= c_u \in \mathsf{num}_1(\hat{a}_u)$. In this case, the bound*

$$\mathsf{m}^u_{\hat{a}_u,a}(s,\psi)\mathsf{cost}(\hat{a}_u) + \mathsf{m}^v_{\hat{a}_u,a}(s,\psi)\mathsf{cost}(a) =$$
$$2\sqrt{\frac{(w_0 - s[v])\mathsf{cost}(a)\mathsf{cost}(\hat{a}_u)}{c_u}} - \frac{c + s[u]}{c_u}\mathsf{cost}(\hat{a}_u)$$

is described via two multiplicators

$$\mathsf{m}^u_{\hat{a}_u,a}(s,\psi) = \sqrt{\frac{(w_0 - s[v])\mathsf{cost}(a)}{c_u \mathsf{cost}(\hat{a}_u)}} - \frac{c + s[u]}{c_u},$$
$$\mathsf{m}^v_{\hat{a}_u,a}(s,\psi) = \sqrt{\frac{(w_0 - s[v])\mathsf{cost}(\hat{a}_u)}{c_u \mathsf{cost}(a)}}.$$

We use $\mathsf{m}^u_{\hat{a}_u,a}(s,\psi) = 0$ if $\mathsf{cost}(\hat{a}_u) = 0$ and $\mathsf{m}^v_{\hat{a}_u,a}(s,\psi) = 0$ if $\mathsf{cost}(a) = 0$.

Note that we use $\mathsf{m}^v_{\hat{a}_u,a}(s,\psi) = 1$ if $\mathsf{cost}(\hat{a}_u) = 0$ in practice since a must be applied at least once. We also use $\mathsf{m}^u_{\hat{a}_u,a}(s,\psi) = 1$ if $\mathsf{cost}(a) = 0$ and $s[u] = 0$ in practice since $\hat{a}_u$ must be applied at least once. To complete the adjustments to obtain the LM-cut heuristic, we define an extension of the function to choose the representative preconditions (pcf') and the corresponding JG. We start with the extension of $h^{\max}_{\text{cri}}(s,\psi)$ for SOSE. As in the original $h^{\max}_{\text{cri}}(s,\psi)$, this heuristic for a given fact corresponds to its distance from the $n_\emptyset$ in the JG. Formally, it is described as recursive equations.

Definition 4 (Precondition Choice Heuristic)**.** *As in all other max heuristics, we set $h^{\max}_{cri,2}(s,\psi) = 0$ for $s \models \psi$, and $h^{\max}_{cri,2}(s,F) = \max_{\psi \in F} h^{\max}_{cri,2}(s,\psi)$ for $F \subseteq \mathcal{F} \cup \Psi^2$. For other conditions $\psi \in \mathcal{F} \cup \Psi^2$ we define $h^{\max}_{cri,2}(s,\psi) = \min\{\bar{h}^{\max}_{cri,1}(s,\psi), \bar{h}^{\max}_{cri,2}(s,\psi)\}$. Where, we take the minimum over the simple version of the $h^{\max}_{cri}$ heuristic*

$$\bar{h}^{\max}_{cri,1}(s,\psi) = \min_{a \in \mathsf{supp}(\psi)} h^{\max}_{cri,2}(s,\mathsf{pre}(a)) + \mathsf{m}_a(s,\psi)\mathsf{cost}(a),$$

and the SOSE achievers, if they exist. I.e., for $\psi : v \geq w_0$

$$\bar{h}^{\max}_{cri,2}(s,\psi) = \min_{\langle \hat{a}_u,a\rangle \in \mathsf{supp}_2(\psi) : \mathsf{m}^u_{\hat{a}_u,a}(s,\psi) > 0} h^{\max}_{cri,2}(s,\mathsf{pre}(a) \cup \mathsf{pre}(\hat{a}_u))$$
$$+ \mathsf{m}^u_{\hat{a}_u,a}(s,\psi)\mathsf{cost}(\hat{a}_u) + \mathsf{m}^v_{\hat{a}_u,a}(s,\psi)\mathsf{cost}(a).$$

If the formula is undefined for ψ we set $\bar{h}^{\max}_{cri,2}(s,\psi) = \infty$.

For each action pair $\langle \hat{a}_u, a\rangle \in \mathsf{supp}_2(\psi)$ we define the precondition choice function as the union of preconditions

$$\mathsf{pcf}(s,\hat{a}_u,a) \in \operatorname*{arg\,max}_{\psi \in \mathsf{pre}(\hat{a}_u) \cup \mathsf{pre}(a)} h^{\max}_{\text{cri},2}(s,\psi).$$

We can finally define the JG required for the cut estimates.

Definition 5 (Justification Graph)**.** *In the justification graph (JG), for each numeric condition $\psi \in \Psi^2$, we create node n_ψ. There is an edge between each two n_ψ and $n_{\psi'}$, if*

$$\{(n_\psi, n_{\psi'}; \langle a\rangle) : a \in \mathsf{supp}(\psi'), \psi = \mathsf{pcf}(s,a)\}, \text{ or}$$
$$\{(n_\psi, n_{\psi'}; \langle \hat{a}_u, a\rangle) : \langle \hat{a}_u, a\rangle \in \mathsf{supp}_2(\psi')$$
$$\psi = \mathsf{pcf}(s,\hat{a}_u,a), \mathsf{m}^u_{\hat{a}_u,a}(s,\psi') > 0\}.$$

The weight function W is defined as

$$(n_\psi, n_{\psi'}; \langle a\rangle) \mapsto \mathsf{m}_a(s,\psi) \cdot \mathsf{cost}(a)$$
$$(n_\psi, n_{\psi'}; \langle \hat{a}_u, a\rangle) \mapsto \mathsf{m}^u_{\hat{a}_u,a}(s,\psi) \cdot \mathsf{cost}(\hat{a}_u) + \mathsf{m}^v_{\hat{a}_u,a}(s,\psi) \cdot \mathsf{cost}(a).$$

For each edge we also define the function lbl that maps edges to their corresponding labels $(n_\psi, n_{\psi'}; \langle x\rangle) \mapsto \langle x\rangle$, where each label can be seen as a set of either one or two action labels. By abuse of notation, we say $a \in \mathsf{lbl}(L)$ if there exists edge e in L such that $a \in \mathsf{lbl}(e)$. Given the JG, we can now show that each cut in the graph that separates the vertices $n_\emptyset$ and n_g where $g \in G$ corresponds to an admissible estimate of an optimal plan.

Theorem 3. *Let Π_{OVC} be the* OVC *of a solvable* LT *with a non-zero optimal cost. Let L be a directed cut in a JG of the second-order delete-relaxation Π^2_{OVC}. For action a in L, let the minimum weight of edges including a be $\mathsf{W}^L(a) = \min_{e \in L : \exists a \in \mathsf{lbl}(e)} \mathsf{W}(e)$ and cost_1 be defined as*

$$\mathsf{cost}_1(a) = \begin{cases} \frac{\mathsf{W}(L)}{\mathsf{W}^L(a)}\mathsf{cost}(a) & \text{if } a \in \mathsf{lbl}(L) \\ 0 & \text{otherwise.} \end{cases}$$

Let $\Pi^2_{\text{OVC},1}$ be a copy of Π^2_{OVC} except that action a has cost $\mathsf{cost}_1(a)$. The weight of the cut $\mathsf{W}(L) = \min_{e \in L} \mathsf{W}(e)$ is admissible for $\Pi^2_{\text{OVC},1}$.

Proof Sketch: Using Thm. 2, we show that

$$\min \left\{ \begin{matrix} \min_{(n_\psi, n_{\psi'}; a) \in L} \mathsf{m}_a(s,\psi') \cdot \mathsf{cost}_1(a), \\ \min_{(n_\psi, n_{\psi'}; \langle \hat{a}_u, a\rangle) \in L} \begin{Bmatrix} \mathsf{m}^u_{\hat{a}_u,a}(s,\psi') \cdot \mathsf{cost}_1(\hat{a}_u) \\ + \mathsf{m}^v_{\hat{a}_u,a}(s,\psi) \cdot \mathsf{cost}_1(a) \end{Bmatrix} \end{matrix} \right\}$$

is a lower bound on the optimal cost. Then, we show that $\mathsf{W}(L)$ is less than or equal to the above lower bound. The inequalities follow directly from the definitions (see the SM). □

We name the resulting heuristic $h_2^{\text{LM-cut}}$. By Thm. 1 in Kuroiwa et al. (2021) we have the admissibility, and by Cor. 1 in the same source we know that it can be computed in polynomial time.

	h^{blind}			$h_{\text{IP}}^{\text{C,prop}}$			h^{irmax}			$h_1^{\text{LM-cut}}$			$h_2^{\text{LM-cut}}$		
All SOSE	C	T	E	C	T	E	C	T	E	C	T	E	C	T	E
FO-COUNT (20)	**4**	5.0	91020	**4**	82.2	199864	**4**	10.6	67523	**4**	4.5	41250	**4**	**2.1**	**9199**
FO-COUNT-INV (20)	3	2.8	66830	3	43.2	131330	3	6.9	62426	3	3.4	43553	**4**	**0.6**	**3464**
FO-COUNT-RND (60)	**14**	9.4	145186	**14**	182.1	344968	**14**	27.3	150918	**14**	7.3	59166	**14**	**0.4**	**1927**
FO-SAILING (20)	2	61.3	2985973	1	828.9	2985973	2	101.6	2882949	2	80.6	2783270	**4**	**44.6**	**1166002**
FO-FARMLAND (50)	14	5.8	280507	12	96.2	439048	14	12.3	280501	14	10.3	280441	**25**	**0.2**	**1018**
Some SOSE															
LIN-CAR-EXP (34)	26	**31.7**	3368599	25	433.2	3368599	**27**	46.1	2796052	**27**	38.3	2795713	**27**	45.2	**2787432**
LIN-CAR-EXP-UNIT (34)	23	**25.1**	2713351	21	351.0	2922774	25	38.4	2207347	25	35.5	2081447	**29**	34.9	**1842544**
No SOSE															
TPP-METRIC (40)	**5**	**1.9**	40406	**5**	86.5	50564	**5**	2.7	23531	**5**	7.1	**11256**	**5**	8.0	11275
ROVER-METRIC (10)	4	4.4	154592	**6**	0.8	**12**	4	0.7	3147	**6**	**0.0**	30	**6**	**0.0**	30
ZENOTRAVEL-LINEAR (10)	4	17.5	393248	7	8.3	**94**	4	3.2	7726	**9**	**0.1**	114	**9**	**0.1**	114
LIN-CAR-POLY (34)	13	**12.5**	1749300	13	225.8	1749300	13	24.7	1508975	**14**	16.7	**1478517**	**14**	17.6	**1478517**
LIN-CAR-POLY-UNIT (34)	13	**12.9**	1776927	13	246.5	1833359	**14**	27.1	1566398	**14**	32.6	**1544265**	**14**	32.5	**1544265**
TOTAL (366)	125	-	-	124	-	-	129	-	-	137	-	-	**155**	-	-

Table 1: Experimental comparison of the admissible heuristics for linear numeric planning. 'C' is the coverage, 'T' is the search time in seconds, and 'E' is the number of expansions. 'T' and 'E' are averaged over instances solved by all methods.

Experimental Evaluation

We experimentally compare $h_1^{\text{LM-cut}}$ and $h_2^{\text{LM-cut}}$. We use the blind heuristic (h^{blind}), the repetition-based max heuristic (h^{irmax}) (Aldinger and Nebel 2017), and the delete-relaxation heuristic ignoring numeric variables ($h_{\text{IP}}^{\text{C,prop}}$) (Piacentini et al. 2018b,a) as baselines. h^{blind} returns 0 if a state is a goal node and $\min_{a \in \mathcal{A}} \mathsf{cost}(a)$ otherwise. h^{irmax} is admissible in general numeric planning including linear numeric planning. These heuristics do not exploit particular structures of linear numeric planning, so our LM-cut heuristics are the first admissible heuristics designed for linear numeric planning. We execute A* (Hart, Nilsson, and Raphael 1968) with these heuristics and evaluate the performance.

We run the experiments on an Intel Xeon Gold 6148 processor with a 30 minute time and 4 GB memory limit using GNU parallel (Tange 2011). The heuristics are implemented in Numeric Fast Downward (Aldinger and Nebel 2017) using C++11 with GCC 9.2, Python 2.7.5, and CPLEX 20.1.0. For $h_1^{\text{LM-cut}}$ and $h_2^{\text{LM-cut}}$, we use redundant constraints in the same way as Scala et al. (2016a).

FO-COUNT, FO-COUNT-INV, FO-COUNT-RND, FO-SAILING, and FO-FARMLAND are the extensions of the SCT domains, where all linear effects are SOSE (Li et al. 2018). Since action costs depend on states in IPC domain TPP-METRIC, which has no SOSE, we use its unit-cost version. ROVER-METRIC and ZENOTRAVEL-LINEAR are introduced by Leofante et al. (2020) and also have no SOSE. The domains prefixed by LIN-CAR- are compiled from PDDL+ domain LIN-CAR (Fox and Long 2006) using a recently proposed method (Percassi, Scala, and Vallati 2021), the details of which are described in the SM. Some linear effects in LIN-CAR-EXP are SOSE, but $\mathsf{cost}(\hat{a}_u) = 0$ for each $\langle \hat{a}_u, a \rangle \in \mathsf{supp}_2(\psi)$, which results in $\mathsf{m}^v_{\hat{a}_u,a}(s, \psi) = 1$ and does not make much difference between $h_1^{\text{LM-cut}}$ and $h_2^{\text{LM-cut}}$. Thus, we also use the unit cost versions suffixed by -UNIT.

We compare coverage, the search time, and the number of expansions in Table 1. $h_2^{\text{LM-cut}}$ has the highest coverage in all domains. In the domains with SOSE, $h_2^{\text{LM-cut}}$ solves more instances than $h_1^{\text{LM-cut}}$ and substantially reduces the search time and the number of expansions, which confirms that our approach successfully exploits the second-order structure. In the domains without SOSE, compared to the baselines, $h_1^{\text{LM-cut}}$ solves more instances in two domains and reduces the number of expansions in three domains. This result shows the benefit of using the LM-cut in linear numeric planning even without SOSE. Note that $h_2^{\text{LM-cut}}$ and $h_1^{\text{LM-cut}}$ are theoretically the same in these domains and the differences come from the randomness of tie-breaking. When goal conditions are all numeric, $h_{\text{IP}}^{\text{C,prop}}$ always returns zero while h^{blind} returns the minimum action-cost for non-goal states, which is why $h_{\text{IP}}^{\text{C,prop}}$ expands more states than h^{blind} in some domains.

Conclusion

Extending the LM-cut heuristic, which is used in classical planning and numeric planning with simple conditions, we proposed the first admissible heuristics for linear numeric planning. In the experiment, both variants perform better than state-of-the-art baselines, and the one using the second-order structure of linear effects has the higher coverage. Theoretically, our heuristics estimate the optimal costs in the two different relaxations of linear numeric planning. Developing tighter heuristics based on the relaxations is a possible direction for future work.

References

Aldinger, J.; Mattmüller, R.; and Göbelbecker, M. 2015. Complexity of interval relaxed numeric planning. In *KI 2015: Advances in Artificial Intelligence*, 19–31.

Aldinger, J.; and Nebel, B. 2017. Interval Based Relaxation Heuristics for Numeric Planning with Action Costs. In *KI 2017: Advances in Artificial Intelligence*, 15–28.

Bonet, B.; and Geffner, H. 2001. Planning as heuristic search. *Artif. Intell.*, 129(1-2): 5–33.

Fox, M.; and Long, D. 2006. Modelling Mixed Discrete-Continuous Domains for Planning. *J. Artif. Intell. Res.*, 27: 235–297.

Hart, P. E.; Nilsson, N. J.; and Raphael, B. 1968. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. Syst. Sci. Cybern.*, 4(2): 100–107.

Helmert, M. 2002. Decidability and Undecidability Results for Planning with Numerical State Variables. In *Proc. AIPS*, 44–53. AAAI.

Helmert, M.; and Domshlak, C. 2009. Landmarks, Critical Paths and Abstractions: What's the Difference Anyway? In *Proc. ICAPS*, 162–169.

Hoffmann, J. 2003. The Metric-FF Planning System: Translating "Ignoring Delete Lists" to Numeric State Variables. *J. Artif. Intell. Res.*, 20: 291–341.

Katz, M.; and Domshlak, C. 2008. Optimal Additive Composition of Abstraction-based Admissible Heuristics. In *Proc. ICAPS*, 174–181.

Keyder, E.; Hoffmann, J.; and Haslum, P. 2012. Semi-relaxed plan heuristics. In *Proc. ICAPS*, 128–136.

Kuroiwa, R.; Shleyfman, A.; and Beck, J. C. 2022. Supplement for LM-cut Heuristics for Optimal Linear Numeric Planning. https://tidel.mie.utoronto.ca/pubs/Kuroiwa_ICAPS2022_supplement.pdf.

Kuroiwa, R.; Shleyfman, A.; Piacentini, C.; Castro, M. P.; and Beck, J. C. 2021. LM-cut and Operator Counting Heuristics for Optimal Numeric Planning with Simple Conditions. In *Proc. ICAPS*, 210–218.

Leofante, F.; Giunchiglia, E.; Ábrahám, E.; and Tacchella, A. 2020. Optimal Planning Modulo Theories. In *Proc. IJCAI*, 4128–4134.

Li, D.; Scala, E.; Haslum, P.; and Bogomolov, S. 2018. Effect-Abstraction Based Relaxation for Linear Numeric Planning. In *Proc. IJCAI*, 4787–4793.

Percassi, F.; Scala, E.; and Vallati, M. 2021. Translations from Discretised PDDL+ to Numeric Planning. In *Proc. ICAPS*, 252–261.

Piacentini, C.; Castro, M. P.; Ciré, A. A.; and Beck, J. C. 2018a. Compiling Optimal Numeric Planning to Mixed Integer Linear Programming. In *Proc. ICAPS*, 383–387.

Piacentini, C.; Castro, M. P.; Ciré, A. A.; and Beck, J. C. 2018b. Linear and Integer Programming-Based Heuristics for Cost-Optimal Numeric Planning. In *Proc. AAAI*, 6254–6261.

Pommerening, F.; Röger, G.; Helmert, M.; and Bonet, B. 2014. LP-Based Heuristics for Cost-Optimal Planning. In *Proc. ICAPS*, 226–234.

Röger, G.; Pommerening, F.; and Helmert, M. 2014. Optimal Planning in the Presence of Conditional Effects: Extending LM-Cut with Context Splitting. In *Proc. ECAI*, 765–770.

Scala, E.; Haslum, P.; Magazzeni, D.; and Thiébaux, S. 2017. Landmarks for Numeric Planning Problems. In *Proc. IJCAI*, 4384–4390.

Scala, E.; Haslum, P.; and Thiébaux, S. 2016. Heuristics for Numeric Planning via Subgoaling. In *Proc. IJCAI*, 3228–3234.

Scala, E.; Haslum, P.; Thiébaux, S.; and Ramírez, M. 2016a. Interval-Based Relaxation for General Numeric Planning. In *Proc. ECAI*, 655–663.

Scala, E.; Haslum, P.; Thiébaux, S.; and Ramírez, M. 2020. Subgoaling Techniques for Satisficing and Optimal Numeric Planning. *J. Artif. Intell. Res.*, 68: 691–752.

Scala, E.; Ramírez, M.; Haslum, P.; and Thiébaux, S. 2016b. Numeric Planning with Disjunctive Global Constraints via SMT. In *Proc. ICAPS*, 276–284.

Tange, O. 2011. GNU Parallel - The Command-Line Power Tool. *;login: The USENIX Magazine*, 36: 42–47.

Yang, F.; Culberson, J.; Holte, R.; Zahavi, U.; and Felner, A. 2008. A General Theory of Additive State Space Abstractions. *J. Artif. Intell. Res.*, 32: 631–662.