Optimal Parity Resolution in Difference Models via Numeric Planning

Luigi Bonassi[1], **James Wilson**[2], **Ruth Chang**[1], **Kit Fine**[3], **Nick Hawes**[1]

[1]University of Oxford, UK
[2]Dyson Institute of Engineering and Technology, UK
[3]New York University, USA
luigibonassi@robots.ox.ac.uk, james.wilson@dyson.com, ruth.chang@law.ox.ac.uk, kit.fine@nyu.edu, nickh@robots.ox.ac.uk

Abstract

AI systems that are designed to automate or support decision making should take into account the fact that human beings are often faced with "hard choices", choices in which neither option is better or equally good as the other. Instead, in hard choices, the options are "on a par", qualitatively different yet in the same overall neighbourhood of value. Difference Models, in which pairs of items are assigned an approximate difference in value, offer a formal framework for representing such relationships and naturally capture cases of parity. They also provide a principled method for resolving parities, thereby establishing a total ranking among the items. A parity resolution can be viewed as the user's commitment to one option over another; and, in this paper, we study the problem of determining the minimal number of such resolutions needed to eliminate all parities. We represent this as the Resolution Number Problem and prove it NP-complete by establishing a correspondence with the Jump Number Problem for partially ordered sets. We then study how to compute a minimal set of resolutions in practice. We show that this problem can be naturally encoded as a numeric planning problem in which optimal plans correspond to optimal resolution sequences. Finally, we evaluate state-of-the-art optimal numeric planners on a set of synthetic Difference Models to assess their scalability on this new class of problems.

Introduction

Making decisions often involves *hard choices*, cases where options are neither better, worse, nor equal, but instead *on a par*. Such situations arise when alternatives differ qualitatively yet lie within the same overall neighbourhood of value (Chang 2017). For example, a business career is qualitatively different from one in academic computer science, and two such careers could be in the same overall neighbourhood of "goodness as a career": the choice between them is hard.

As autonomous and AI-driven systems become increasingly prevalent in society, approaches that account for hard choices are vital. From resource and task allocation (Zahedi, Sengupta, and Kambhampati 2024) to personal decision support, systems that fail to recognize or properly deal with parity risk making decisions that might be misaligned with user priorities. The consequences of these limitations are already evident: discriminatory outcomes in machine-learning CV screening, biases in facial recognition, and more. Despite these issues, users often over-trust existing systems. For instance, seeking advice from Large Language Models (LLMs) on medical decisions (Yun and Bickmore 2025), or even life-or-death military action (Holbrook et al. 2024), unaware that these models merely aim to produce plausible responses, and do not guarantee truth, impartiality, relevance, or alignment with personal values (Bender et al. 2021).

To earn trust, we believe AI systems should serve user interests through verifiable mathematical methods (Abeywickrama et al. 2023) that capture the diverse preferences and decision factors behind any recommendation, plan, or choice.

In this paper, we focus on *Difference Models*, a framework that represents parity by assigning approximate value differences to item pairs, and offers an effective *resolution* method to eliminate such parities. From a user's perspective, parity arises when an object is neither better nor equal to another, and a resolution corresponds to committing to one of the options. Furthermore, we assume that users aim to eliminate all parities with the fewest commitments and address a specific challenge within this setting: determining the *minimum* number of resolutions to eliminate all parities, thus providing a measure of the minimal effort to reach this objective. Specifically, we formally prove that the decision version of this problem is NP-complete. We then present an encoding of this problem to numeric planning (Fox and Long 2003), enabling the use of established planning techniques to compute a minimal set of resolutions. This encoding addresses key combinatorial challenges arising when minimising user commitments, and lays the groundwork for more advanced parity-aware decision-making approaches. Finally, we empirically evaluate our approach to assess the performance of state-of-the-art numeric planners under this encoding.

Related Work

This paper focuses on modelling choices between items and uses planning to address the computational challenges arising in this setting. Some interpret hard choices as "uncertainties" requiring more data, or "indeterminacies" allowing for arbitrary resolution (Dobbe, Gilbert, and Mintz 2021; McElfresh et al. 2021). In other scenarios, decision-making systems rely on optimization-based techniques that rank options by aggregating preferences (Noothigattu et al. 2018). However, these approaches are limited in contexts where trade-offs between competing values cannot be reduced to a single metric. We therefore consider models that explicitly represent parities and incorporate mechanisms to refine them.

Recent work in robotics acknowledges the need to recognise competing trade-offs and seek user clarification rather than assuming a single optimal outcome. This challenge has motivated strategies that avoid committing to a single plan, such as multi-objective planning frameworks which balance completing objectives under uncertainty (Stephens et al. 2024), or multi-dimensional parameter sets translated to user-understandable trade-offs to ease interaction with cognitively demanding systems (Wilson and Hauert 2022). Related works also deal with the broad problem of human-aware planning (Zahedi, Sengupta, and Kambhampati 2024; Chakraborti, Sreedharan, and Kambhampati 2018), human-robot interaction (Nguyen et al. 2025), and integrating human's trust in the planning process (Zahedi et al. 2023).

Like difference models, partially ordered sets (Davey and Priestley 2002) and related ordered structures have been considered to reason about incomparable objects. For example, semiorders (Luce 1956) model human preferences by assigning numerical utilities to objects and treating as incomparable any pair whose utilities differ by less than a fixed threshold. Other alternatives to semiorders have also been explored, such as interval orders (Fishburn 1970) and bounded bitolerance orders (Bogart and Trenk 1994).

Background: Difference Models

This section introduces the notion of a Difference Model, or D-Model (Fine 2026), which is a structure that relates pairs of objects in terms of their difference in value. As this value difference may be imprecise, D-Models use the notion of *approximate difference*. This is a tuple $\langle r, s\rangle$, representing an interval, with $r, s \in \mathbb{Z}$ and $r \leqslant s$. Given a set of items X, between which a choice is to be made, we denote by "$\dot{-}$" a function mapping two items $y, x \in X$ into an approximate difference $\langle r, s\rangle$. We might think of an approximate difference as approximating to the real difference in value between two items. Thus, $(y \dot{-} x) = \langle 1, 3\rangle$ indicates that the "real" difference in value between y and x is at least 1 and at most 3.[1] If $r = s$, then $(y \dot{-} x)$ is a precise difference.

Within this framework, we can determine whether an item is better than another by analysing their approximate difference. To do so, we first define the following relations between two approximate differences $\langle r_0, s_0\rangle$ and $\langle r_1, s_1\rangle$:

$\langle r_0, s_0\rangle \geqslant \langle r_1, s_1\rangle$ iff $r_0 \geqslant r_1$ and $s_0 \geqslant s_1$

$\langle r_0, s_0\rangle > \langle r_1, s_1\rangle$ iff ($r_0 > r_1$ and $s_0 \geqslant s_1$) or ($r_0 \geqslant r_1$ and $s_0 > s_1$)

$\langle r_0, s_0\rangle \approx \langle r_1, s_1\rangle$ iff neither $\langle r_0, s_0\rangle \geqslant \langle r_1, s_1\rangle$ nor $\langle r_0, s_0\rangle \leqslant \langle r_1, s_1\rangle$

For example, $\langle 4, 6\rangle \geqslant \langle 3, 5\rangle$, and also $\langle 4, 6\rangle > \langle 3, 5\rangle$. Moreover, $\langle 4, 6\rangle \approx \langle -2, 7\rangle$ as neither approximate difference is greater than or equal to the other. We now define the following relations between two items $y, x \in X$:

$x \prec y$ iff $(y \dot{-} x) > \langle 0, 0\rangle$ (better)

$y \sim x$ iff $(y \dot{-} x) = \langle 0, 0\rangle$ (equal)

$x \preceq y$ iff either $x \prec y$ or $y \sim x$ (better or equal)

$y \approx x$ iff $(y \dot{-} x) \approx \langle 0, 0\rangle$. (on a par)

Note that y and x are on a par when neither is better nor are they equal.

Example 1. *As a running example, we model the difference in value between career paths. Let A indicate a career in arts, and B a career in* business. *If $(A \dot{-} B) = \langle 1, 3\rangle$, then $B \prec A$, since $\langle 1, 3\rangle > \langle 0, 0\rangle$. Intuitively, $\langle 1, 3\rangle$ indicates that the difference between A and B must be positive, meaning that the arts career is strictly better. On the other hand, if $(A \dot{-} B) = \langle -2, 3\rangle$ then $A \approx B$, since $\langle -2, 3\rangle \approx \langle 0, 0\rangle$. Intuitively, this indicates that the difference between A and B could be taken to be positive (favouring arts) or negative (favouring business). Since the two items are not of equal value, and neither is better, they are on a par.*

Given $(y \dot{-} x) = \langle r, s\rangle$, we use $(y \dot{-} x)\!\downarrow$ to indicate r and $(y \dot{-} x)\!\uparrow$ to indicate s. Note that $x \prec y$ when $(y \dot{-} x)\!\downarrow \geqslant 0$ and $(y \dot{-} x)\!\uparrow > 0$, $y \sim x$ when $(y \dot{-} x)\!\downarrow = (y \dot{-} x)\!\uparrow = 0$, and $y \approx x$ when $(y \dot{-} x)\!\downarrow < 0$ and $(y \dot{-} x)\!\uparrow > 0$. In addition, we define the following relation and operators for two approximate differences $\langle r_0, s_0\rangle$ and $\langle r_1, s_1\rangle$:

$$\langle r_0, s_0\rangle \subseteq \langle r_1, s_1\rangle \text{ iff } r_0 \geqslant r_1 \text{ and } s_0 \leqslant s_1$$
$$\langle r_0, s_0\rangle + \langle r_1, s_1\rangle = \langle r_0 + r_1, s_0 + s_1\rangle$$
$$-\langle r_0, s_0\rangle = \langle -s_0, -r_0\rangle$$

The $\subseteq$ relation indicates that one approximate difference is contained within another. For example, $\langle 4, 6\rangle \subseteq \langle -2, 7\rangle$. Moreover, summing two approximate differences is done by pairwise summing both end-points, and negation flips both the signs and the endpoints of the interval. For example, $\langle 4, 6\rangle + \langle -2, 7\rangle = \langle 2, 13\rangle$ and $-\langle -2, 7\rangle = \langle -7, 2\rangle$.

A D-Model is a "coherent" collection of approximate differences over a given set of items.

Definition 1. *A D-Model is a tuple $M = \langle X, \dot{-}\rangle$, where X is a non-empty set and $\dot{-}$ is a function mapping two items $y, x \in X$ into an approximate difference. Moreover, for every $x, y, z \in X$, we require:*

$(x \dot{-} x) = \langle 0, 0\rangle$ (Equality)

$(y \dot{-} x) = -(x \dot{-} y)$ (Reversal)

$(z \dot{-} x) \subseteq (z \dot{-} y) + (y \dot{-} x)$. (Triangular Inequality)

Equality requires each item to be of equal value to itself, whilst reversal requires $(y \dot{-} x)$ to be the negative of $(x \dot{-} y)$. For this reason, $y \sim x$ iff $x \sim y$ and $x \approx y$ iff $y \approx x$. Triangular inequality requires that the minimum (maximum, resp.) difference between z and x be compatible with the sum of the minimum (maximum, resp.) difference between z and y and the minimum (maximum, resp.) difference between y and x. Indeed, supposing $(z \dot{-} x) = \langle r_0, s_0\rangle, (z \dot{-} y) = \langle r_1, s_1\rangle, (y \dot{-} x) = \langle r_2, s_2\rangle$, triangular inequality then requires $\langle r_0, s_0\rangle \subseteq \langle r_1 + r_2, s_1 + s_2\rangle$ which is to say that $r_0 \geqslant r_1 + r_2$ and $s_0 \leqslant s_1 + s_2$.

The next example illustrates an entire D-Model.

[1] Under a more objective understanding, $\langle 1, 3\rangle$ *is* the real difference between the items, and it is a mistake to suppose, without the user making a further commitment, that there is a more precise objective difference in value between them (Chang 2002).

Example 2. *As in Example 1, let A indicate a career in arts and B a career in* business. *Let also C be a career in* computer science. *We define the D-Model $M = \langle X, \dot{-} \rangle$ in the table below:*

$\dot{-}$	A	B	C
A	$\langle 0,0 \rangle$	$\langle -2,2 \rangle$	$\langle 0,1 \rangle$
B	$\langle -2,2 \rangle$	$\langle 0,0 \rangle$	$\langle -2,3 \rangle$
C	$\langle -1,0 \rangle$	$\langle -3,2 \rangle$	$\langle 0,0 \rangle$

Under this tabular notation, the set of items X is given by the set $X = \{A, B, C\}$ of row and column labels. For each pair of items, the entry at row x and column y gives $(x \dot{-} y)$. For example, $(A \dot{-} B) = \langle -2, 2 \rangle$. The function $\dot{-}$ satisfies equality, since $(A \dot{-} A) = (B \dot{-} B) = (C \dot{-} C) = \langle 0, 0 \rangle$ "along the diagonal". It also satisfies reversal, with, for example, $(B \dot{-} C) = \langle -2, 3 \rangle$ and $(C \dot{-} B) = -(B \dot{-} C) = \langle -3, 2 \rangle$. Finally, triangular inequality can be seen to hold. For example, $(A \dot{-} B) \subseteq (B \dot{-} C) + (C \dot{-} A)$, since $\langle -2, 2 \rangle \subseteq \langle -2, 3 \rangle + \langle -1, 0 \rangle$.

It can be shown that triangular inequality implies that the relation $\preceq$ in a difference model is transitive.

Theorem 1. *Let $M = \langle X, \dot{-} \rangle$ be a D-Model. For any $x, y, z \in X$, if $x \preceq y$ and $y \preceq z$ then $x \preceq z$.*

Proof. Let $(y \dot{-} x) = \langle r_1, s_1 \rangle$ and $(z \dot{-} y) = \langle r_2, s_2 \rangle$. Since $x \preceq y$ and $y \preceq z$, $r_1 \geqslant 0$, $s_1 \geqslant 0$, $r_2 \geqslant 0$, $s_2 \geqslant 0$. By triangular inequality: $(z \dot{-} x) \subseteq (z \dot{-} y) + (y \dot{-} x) = \langle r_1 + r_2, s_1 + s_2 \rangle$. But note that $r_1 + r_2 \geqslant 0$ and $s_1 + s_2 \geqslant 0$. If $r_1 + r_2 = 0$ and $s_1 + s_2 = 0$, then $(z \dot{-} x) \subseteq \langle 0, 0 \rangle$ and so $z \sim x$. Otherwise, $x \prec z$. In either case, $x \preceq z$. □

We now introduce the concept of a *resolution*, which we regard as a way of modifying a parity within a D-Model so as to make one item better than the other. More exactly, suppose given a D-Model $M = \langle X, \dot{-} \rangle$ and a parity within M, i.e. a pair of items $u, v \in X$ for which $u \approx v$. The next definition then shows how to compute a new D-Model in which the parity is resolved in favor of v being better than u.

Definition 2 (Resolution)**.** *Let $M = \langle X, \dot{-} \rangle$ be a D-Model and $u, v \in X$ two items for which $u \approx v$. We set $M[u, v] = \langle X, \dot{-}' \rangle$ where, for every $y, x \in X$, the value $(y \dot{-}' x) = \langle r*, s* \rangle$ of the new function $\dot{-}'$ is defined by:*

$$r^* = \max((y \dot{-} x)\downarrow, (y \dot{-} v)\downarrow + (u \dot{-} x)\downarrow)$$
$$s^* = \min((y \dot{-} x)\uparrow, (y \dot{-} u)\uparrow + (v \dot{-} x)\uparrow)$$

Recall that, in any D-Model M, $u \approx v$ iff $(v \dot{-} u)\downarrow < 0$ and $(v \dot{-} u)\uparrow > 0$. Intuitively, the minimal modification required to make $u \prec v$ would be to set $(v \dot{-}' u) = \langle 0, (v \dot{-} u)\uparrow \rangle$. Indeed, if we replace y with v and u with x in Definition 2, then we obtain exactly this result. In addition, every approximate difference $(y \dot{-} x)$ will be transformed into a possibly smaller one: for, by Definition 2, r^* is always greater than or equal to the previous lower bound, whilst s^* is always smaller than or equal to the previous upper bound. This adjustment is necessary in order to maintain triangular inequality, since simply setting $(v \dot{-}' u)$ to $\langle 0, (v \dot{-} u)\uparrow \rangle$ would in general lead to its failure.

Given two distinct D-Models $\langle X, \dot{-} \rangle$ and $\langle X, \dot{-}' \rangle$, we hereafter use the notation $\prec'$, $\approx'$, etc., to indicate a relation with respect to $\dot{-}'$ and $\prec$, $\approx$, etc., to indicate a relation with respect to $\dot{-}$.

Theorem 2 (Fine (2026))**.** *Let $M = \langle X, \dot{-} \rangle$ be a D-Model and $u, v \in X$ two items for which $u \approx v$. Then $M[u, v] = \langle X, \dot{-}' \rangle$ is a D-Model in which $u \prec' v$.*

A proof for Theorem 2 is provided in the appendix. The next example shows how resolution is performed.

Example 3. *Let M be the D-Model given in example 2. Since $A \approx B$ in M, we can apply Definition 2 to obtain the new D-Model $M[B, A] = \langle X, \dot{-}' \rangle$ defined by:*

$\dot{-}'$	A	B	C
A	$\langle 0,0 \rangle$	$\langle 0,2 \rangle$	$\langle 0,1 \rangle$
B	$\langle -2,0 \rangle$	$\langle 0,0 \rangle$	$\langle -2,1 \rangle$
C	$\langle -1,0 \rangle$	$\langle -1,2 \rangle$	$\langle 0,0 \rangle$

As discussed, we obtain $(A \dot{-}' B) = \langle 0, (A \dot{-} B)\uparrow \rangle = \langle 0, 2 \rangle$, and thus $B \prec A$. Note also that the value of $(B \dot{-}' C)$ has been modified. Recall that we must have $(B \dot{-}' C) \subseteq (B \dot{-}' A) + (A \dot{-}' C)$. Before, in M, $(B \dot{-} A)$ was $\langle -2, 2 \rangle$, but now $(B \dot{-}' A)$ is $\langle -2, 0 \rangle$, whilst $(A \dot{-}' C)$ is unchanged. Therefore, we must have $(B \dot{-}' C) \subseteq \langle -2, 0 \rangle + \langle 0, 1 \rangle = \langle -2, 1 \rangle$ and thus the resolution restricts $(B \dot{-}' C)$ to $\langle -2, 1 \rangle$ to maintain triangular inequality.

The next section focuses on the problem of removing all of the parities in a D-Model by means of resolution.

The Resolution Number Problem

From Theorem 2, if $u \approx v$ in a D-Model M, then we can obtain a new D-Model $M[u, v]$ in which $u \prec v$. We can therefore "resolve" all parities in a finite D-Model by performing a series of appropriate resolutions. This section formalises the problem of finding a minimal set of such resolutions.

We begin with some definitions. A *resolution sequence* for a D-Model $M = \langle X, \dot{-} \rangle$ is a sequence $r = \langle (u_0, v_0), \ldots, (u_n, v_n) \rangle$ in which each $(u_i, v_i) \in X \times X$. Moreover, if $u \approx v$ does not hold for some u and v in a D-Model M, then $M[u, v] = M$. That is, a resolution for two items that are not on a par does not modify the D-Model. Let $M_0 = M$, $M_1 = M_0[u_0, v_0]$, …, $M_{n+1} = M_n[u_n, v_n]$. We say that the resolution sequence r is *sound* if $\forall i = 0, 1, \ldots, n \cdot u_i \approx v_i$ in M_i. In other words, a sound resolution sequence induces a sequence of D-Models $M_0 = M$, M_1, …, M_{n+1} where $\forall i = 0, 1, \ldots, n$ we have $M_i \neq M_{i+1}$.

For convenience, we will use r_i with $i \leqslant n$ to indicate the subsequence $r_i = \langle (u_0, v_0), \ldots, (u_i, v_i) \rangle$ of r. In addition, where r is *any* resolution sequence for a D-Model M, we use $M[r]$ for the model which results from executing r on M. Note that if r is the empty sequence, then $M[r] = M$.

We say that the resolution sequence r *resolves* M if r is sound and $M[r] = M_{n+1}$ contains no parities. Finally, let $c(r)$ be the number of ordered pairs in the sequence r. We say that r is *optimal* if it has minimal size. We also use $c(M)$ to indicate the size of the optimal resolution sequence for M, that is, $c(M) = \min(\{c(r) \mid r \text{ resolves } M\})$.

Example 4. *Consider the D-Model M in Example 2. The resolution sequence $r = \langle (B, A), (C, B) \rangle$ is sound because $A \approx B$ in M and $B \approx C$ in $M[B, A]$ (this D-Model is reported in Example 3). Moreover, r induces the sequence of D-Models M_0, M_1, M_2, where $M_0 = M$, M_1 is $M[B, A]$ and M_2 is $M[B, A][C, B] = \langle X, \dot{-}'' \rangle$, which is defined by the following table:*

$\dot{-}''$	A	B	C
A	$\langle 0, 0 \rangle$	$\langle 0, 1 \rangle$	$\langle 0, 1 \rangle$
B	$\langle -1, 0 \rangle$	$\langle 0, 0 \rangle$	$\langle 0, 1 \rangle$
C	$\langle -1, 0 \rangle$	$\langle -1, 0 \rangle$	$\langle 0, 0 \rangle$

Note that M_2 has no parities, and so r resolves M. However, r is not optimal, as the shorter resolution sequence $r^* = \langle (A, B) \rangle$ produces a D-Model without parities. Indeed, $c(M)$ is 1 as $M[A, B] = \langle X, \dot{-}^* \rangle$ is given by:

$\dot{-}^*$	A	B	C
A	$\langle 0, 0 \rangle$	$\langle -2, 0 \rangle$	$\langle 0, 1 \rangle$
B	$\langle 0, 2 \rangle$	$\langle 0, 0 \rangle$	$\langle 0, 3 \rangle$
C	$\langle -1, 0 \rangle$	$\langle -3, 0 \rangle$	$\langle 0, 0 \rangle$

We now introduce the *resolution number problem*.

Definition 3 (Resolution Number Problem)**.** *Instance: a D-Model $M = \langle X, \dot{-} \rangle$ and integer $k \geqslant 0$. Question: is $c(M) \leqslant k$?*

In the next sections, we shall show that this problem is NP-complete. For purposes of reduction, we use the closely related *jump number problem* (Kierstead 1986) for partially ordered sets.

Jump Number Problem for Partially Ordered Sets

We briefly introduce partially ordered sets (poset) and state the jump number problem. A poset is a pair $P = \langle X, \leqslant_p \rangle$ where X is a set of objects and $\leqslant_p$ is a binary relation on X that is reflexive, antisymmetric, and transitive. For all $x, y, z \in X$, reflexive means $x \leqslant_p x$, antisymmetric means that if $x \leqslant_p y$ and $y \leqslant_p x$ then $x = y$, and transitive means that if $x \leqslant_p y$ and $y \leqslant_p z$ then $x \leqslant_p z$. We also use $x \nleqslant_p y$ to indicate that $x \leqslant_p y$ does not hold, $x <_p y$ for $x \leqslant_p y$ and $x \neq y$, and finally $x \mid_p y$ to indicate that x and y are incomparable, that is, $x \nleqslant_p y$ and $y \nleqslant_p x$.

We now introduce the notion of *linear extensions*, which are at the core of the jump number problem. Formally, a total order $t = x_1 \leqslant_t x_2 \cdots \leqslant_t x_n$ of all elements of X is a linear extension of the poset $P = \langle X, \leqslant_p \rangle$ if for all $y, x \in X$, $x \leqslant_p y$ implies $x \leqslant_t y$. For a poset P and a linear extension t of P, a *jump* is a consecutive pair of elements x_i, x_{i+1} in t that are incomparable in P (i.e., $x_i \mid_p x_{i+1}$). Let $s(P, t)$ be the number of jumps in t. The *jump number of* P, denoted by $s(P)$, is the minimum number of jumps across all of the linear extensions of P. Formally,

$$s(P) = \min(\{s(P, t') \mid t' \text{ is a linear extension of } P\}).$$

Example 5. *Let $P = \langle \{A, B, C, D\}, \leqslant_p \rangle$ where $B \leqslant_p A$ and $D \leqslant_p C$. The total order $t_0 = A \leqslant_{t_0} B \leqslant_{t_0} C \leqslant_{t_0} D$ is not a linear extension of P, as $B \leqslant_p A$. However, $t_1 = B \leqslant_{t_1} D \leqslant_{t_1} A \leqslant_{t_1} C$ is a valid linear extension of P with 3 jumps: (B, D), (D, A), (A, C). This is because these pairs of items are adjacent in t_1 and incomparable in P. Finally, the jump number $s(P)$ of P is 1, since the linear extension $t_2 = B \leqslant_{t_2} A \leqslant_{t_2} D \leqslant_{t_2} C$ has exactly one jump, (A, D), and there are no other linear extensions with fewer jumps.*

We can now state the *Jump Number Problem* (Colbourn and Pulleyblank 1985; Kierstead 1986).

Definition 4 (Jump Number Problem)**.** *Instance: a poset P and an integer $k \geqslant 0$. Question: is $s(P) \leqslant k$?*

The jump number problem, which asks if the jump number of a poset is lower than or equal to an integer k, is known to be NP-complete (see also (Bouchitté and Habib 1987)).

Computational Complexity

We now establish that the Resolution Number Problem is also NP-complete. The proof is organised as follows. First, we show that every poset can be encoded as a D-Model **(Theorem 3)**. Second, we extend the notion of linear extension and jumps to D-Models, and show how the resolution changes the subclass of D-Models that encode posets **(Theorem 4)**. Third, we show that the jump number of a poset corresponds to the optimal number of resolutions of the corresponding D-Model **(Theorem 5)**.

Theorem 3. *Let $P = \langle X, \leqslant_p \rangle$ be a poset and let $\mathcal{P} = \langle X, \dot{-} \rangle$ where $\dot{-}$ is defined by:*

$$y \dot{-} x = \begin{cases} \langle 0, 0 \rangle & \text{if } x = y \\ \langle 0, 1 \rangle & \text{if } x <_p y \\ \langle -1, 0 \rangle & \text{if } y <_p x \\ \langle -1, 1 \rangle & \text{if } y \mid_p x. \end{cases}$$

Then $\mathcal{P}$ is a D-Model.

Proof. To show that $\mathcal{P}$ is a D-Model, we need to prove that equality, reversal, and triangular inequality hold in $\langle X, \dot{-} \rangle$.
Equality. We can see that $(x \dot{-} x) = \langle 0, 0 \rangle$ for every $x \in X$.
Reversal. For every $y, x \in X$, if $x = y$, we have $(y \dot{-} x) = -(x \dot{-} y) = \langle 0, 0 \rangle$. Otherwise, if $(y \dot{-} x) = \langle 0, 1 \rangle$, then $x <_p y$ and thus, by the definition of $\mathcal{P}$, $(x \dot{-} y) = \langle -1, 0 \rangle$. Similarly, if $(y \dot{-} x) = \langle -1, 0 \rangle$, then $y <_p x$ and thus $(x \dot{-} y) = \langle 0, 1 \rangle$. Lastly, if $(y \dot{-} x) = \langle -1, 1 \rangle$, then $y|_p x$. Since $y|_p x$ implies $x|_p y$, we have $(x \dot{-} y) = \langle -1, 1 \rangle$.
Triangular inequality. To prove this property, we consider every possible values of $(z \dot{-} y)$ and $(y \dot{-} x)$, and show that $(z \dot{-} x) \subseteq (z \dot{-} y) + (y \dot{-} x)$ always holds. First, if $\langle -1, 1 \rangle \subseteq (z \dot{-} y) + (y \dot{-} x)$ then triangular inequality holds because by definition $(z \dot{-} x) \subseteq \langle -1, 1 \rangle$. This happens, for example, when $(z \dot{-} y) = \langle -1, 0 \rangle$ and $(y \dot{-} x) = \langle 0, 1 \rangle$. All other cases are reported in Table 1. As we can see, $(z \dot{-} x) \subseteq (z \dot{-} y) + (y \dot{-} x)$ holds in all cases.

□

Example 6. *Consider the poset P defined in Example 5. Recall that $B \leqslant_p A$ and $D \leqslant_p C$. So, the corresponding D-Model $\mathcal{P}$ according to Theorem 3 is defined by:*

$\dot{-}$	A	B	C	D
A	$\langle 0, 0 \rangle$	$\langle 0, 1 \rangle$	$\langle -1, 1 \rangle$	$\langle -1, 1 \rangle$
B	$\langle -1, 0 \rangle$	$\langle 0, 0 \rangle$	$\langle -1, 1 \rangle$	$\langle -1, 1 \rangle$
C	$\langle -1, 1 \rangle$	$\langle -1, 1 \rangle$	$\langle 0, 0 \rangle$	$\langle 0, 1 \rangle$
D	$\langle -1, 1 \rangle$	$\langle -1, 1 \rangle$	$\langle -1, 0 \rangle$	$\langle 0, 0 \rangle$

$(z \dot{-} y)$	$(y \dot{-} x)$	$(z \dot{-} x)$	Justification
$\langle 0,0\rangle$	$\langle 0,0\rangle$	$\langle 0,0\rangle$	$z = y = x$.
$\langle 0,0\rangle$	$\langle 0,1\rangle$	$\langle 0,1\rangle$	$z = y$ and $x <_p y$ imply $x <_p z$.
$\langle 0,1\rangle$	$\langle 0,0\rangle$	$\langle 0,1\rangle$	Symmetric to previous case.
$\langle 0,1\rangle$	$\langle 0,1\rangle$	$\langle 0,1\rangle$	$x <_p y <_p z$ implies $x <_p z$.
$\langle 0,0\rangle$	$\langle -1,0\rangle$	$\langle -1,0\rangle$	$z = y$ and $y <_p x$ imply $z <_p x$.
$\langle -1,0\rangle$	$\langle 0,0\rangle$	$\langle -1,0\rangle$	Symmetric to previous case.
$\langle -1,0\rangle$	$\langle -1,0\rangle$	$\langle -1,0\rangle$	$z <_p y <_p x$ implies $z <_p x$.

Table 1: Possible values of $(z \dot{-} x)$ based on the values of $(z \dot{-} y)$ and $(y \dot{-} x)$.

As one can see, $B \prec A$ and $D \prec C$. Moreover, a pair of items is incomparable in P iff they are on a par in the corresponding D-Model $\mathcal{P}$.

In giving the rest of the proof, we shall use $\mathcal{P}$ to denote the D-Model obtained from the posets P. Moreover, observe that there is a one to one map between the relations of P and the corresponding relations of $\mathcal{P}$.

Lemma 1. *Let $P = \langle X, \leqslant_p\rangle$ be a poset and $\mathcal{P} = \langle X, \dot{-}\rangle$ be its corresponding D-Model. Then for every $y, x \in X$ the following relations hold:*

- *$y <_p x$ ($y \leqslant_p x$, resp.) iff $y \prec x$ ($y \preceq x$, resp.)*
- *$y \,|_p\, x$ iff $y \approx x$.*

Proof. Full proof in the appendix. □

We now extend the notions of linear extensions and jumps to D-Models. A total order $t = x_1 \leqslant_t x_2 \cdots \leqslant_t x_n$ on X is a linear extension of the D-Model $M = \langle X, \dot{-}\rangle$ if for all $y, x \in X$, $x \preceq y$ in M implies $x \leqslant_t y$. Moreover, a *jump* for M is a consecutive pair of objects x_i, x_{i+1} in t such that $x_i \approx x_{i+1}$. Lemma 1 implies that a poset P and its corresponding D-Model $\mathcal{P}$ share the same linear extensions. Moreover, a jump for P in a linear extension t is also a jump for $\mathcal{P}$, and vice versa.

Example 7. *Consider the poset P defined in Example 5 and its corresponding D-Model $\mathcal{P}$ reported in Example 6. Clearly, the linear extension $t = B \leqslant_t A \leqslant_t D \leqslant_t C$ of P is a linear extension of $\mathcal{P}$. This is because we have both $B \preceq A$ and $D \preceq C$ in $\mathcal{P}$. Moreover, the only jump (A, D) for P is also a jump for $\mathcal{P}$, as $A \approx D$.*

One of the main results of this section is to show that $s(P) = c(\mathcal{P})$, that is, that the jump number of P is equal to the optimal number of resolutions required to resolve $\mathcal{P}$. The case where P has no incomparable items is trivial, as $\mathcal{P}$ then has no parities; and so, in this case, $s(P) = c(\mathcal{P}) = 0$. Therefore, without any loss of generality, we may, for the rest of this section, assume that a poset P has at least one pair of incomparable items and that, consequently, its corresponding D-Model $\mathcal{P}$ has at least one parity.

We now analyse how a resolution changes a D-Model $\mathcal{P}$ encoding a poset $P = \langle X, \leqslant_p\rangle$. It is well known that, if u and v are incomparable objects in P, then we can construct a new poset $Q = \langle X, \leqslant_q\rangle$ with $u \leqslant_q v$ by using the following rule: for every $y, x \in X$, set $x \leqslant_q y$ if either $x \leqslant_p y$ or $x \leqslant_p u$ and $v \leqslant_p y$ (Szpilrajn 1930). The next theorem shows that a resolution performs the corresponding transformation on the D-Model $\mathcal{P}$.

Theorem 4. *Let $P = \langle X, \leqslant_p\rangle$ be a poset, and $\mathcal{P} = \langle X, \dot{-}\rangle$ its corresponding D-Model. If $u \approx v$ and $\mathcal{P}[u, v] = \langle X, \dot{-}'\rangle$, then $\forall y, x \in X \cdot x \preceq' y$ iff either $x \preceq y$ or ($x \preceq u$ and $v \preceq y$). Moreover, $\forall y, x \in X \cdot x \sim y$ iff $x \sim' y$.*

Proof. We first prove that for every $y, x \in X$:

$$x \preceq' y \text{ iff } (x \preceq y) \vee (x \preceq u \wedge v \preceq y).$$

We analyse both directions.
($\Rightarrow$). Suppose $(x \preceq' y)$ but $\neg((x \preceq y) \vee (x \preceq u \wedge v \preceq y))$. Therefore, $(x \npreceq y) \wedge (x \npreceq u \vee v \npreceq y)$. We have two cases: either $(x \npreceq y \wedge x \npreceq u)$ or $(x \npreceq y \wedge v \npreceq y)$. Suppose $(x \npreceq y \wedge x \npreceq u)$. This implies $y \prec x$ or $y \approx x$, and $u \prec x$ or $u \approx x$. In turn, this implies $(y \dot{-} x)\!\downarrow = -1$ and $(u \dot{-} x)\!\downarrow = -1$. In addition, $(y \dot{-} v)\!\downarrow$ can be 0 or -1 (see Theorem 3). Therefore, by Definition 2 we have:

$$(y \dot{-}' x)\!\downarrow = \max((y \dot{-} x)\!\downarrow, (y \dot{-} v)\!\downarrow + (u \dot{-} x)\!\downarrow) = \max(-1, (y \dot{-} v)\!\downarrow + -1) = -1.$$

Therefore $y \prec' x$ or $y \approx' x$ (contradiction). The other case, where $(x \npreceq y \wedge v \npreceq y)$, is analogous.
($\Leftarrow$). Suppose $(x \preceq y) \vee (x \preceq u \wedge v \preceq y)$ but $x \npreceq' y$. This implies $y \prec' x$ or $y \approx' x$. In both cases, $(y \dot{-}' x)\!\downarrow = -1$. Recall that $(y \dot{-}' x)\!\downarrow = \max((y \dot{-} x)\!\downarrow, (y \dot{-} v)\!\downarrow + (u \dot{-} x)\!\downarrow)$. As we are taking the maximum between those two values, if $(y \dot{-}' x)\!\downarrow$ is -1 then both $(y \dot{-} v)\!\downarrow + (u \dot{-} x)\!\downarrow$ and $(y \dot{-} x)\!\downarrow$ must be lower than or equal to -1. Therefore, we have:

1. $(y \dot{-} x)\!\downarrow \leqslant -1$, which implies either $y \prec x$ or $y \approx x$, and thus $x \npreceq y$.
2. $(y \dot{-} v)\!\downarrow + (u \dot{-} x)\!\downarrow \leqslant -1$. Since the lower values of these approximate differences are 0 or -1, this implies that $(y \dot{-} v)\!\downarrow = -1$ or $(u \dot{-} x)\!\downarrow = -1$. If $(y \dot{-} v)\!\downarrow = -1$, we have $y \approx v$ or $y \prec v$. Either way, $v \npreceq y$. Similarly, if $(u \dot{-} x)\!\downarrow = -1$ then $u \approx x$ or $u \prec x$, and thus $x \npreceq u$.

We have that $(x \npreceq y) \wedge (x \npreceq u \vee v \npreceq y)$ (contradiction).
We now show that $y \sim x$ iff $y \sim' x$. ($\Rightarrow$) Immediate. ($\Leftarrow$) Suppose $x \sim' y$ but $x \nsim y$. That is, $(y \dot{-}' x) = \langle 0, 0\rangle$ and $(y \dot{-} x) \neq \langle 0, 0\rangle$. By Definition 2, $(y \dot{-}' x) = \langle 0, 0\rangle$ implies:

$$\max((y \dot{-} x)\!\downarrow, (y \dot{-} v)\!\downarrow + (u \dot{-} x)\!\downarrow) = 0$$
$$\min((y \dot{-} x)\!\uparrow, (y \dot{-} u)\!\uparrow + (v \dot{-} x)\!\uparrow) = 0.$$

Since $(y \dot{-} x) \neq \langle 0, 0\rangle$, we have three possible cases:

- $(y \dot{-} x)\!\downarrow = 0$ and $(y \dot{-} u)\!\uparrow + (v \dot{-} x)\!\uparrow = 0$. By Theorem 3, $(y \dot{-} u)\!\uparrow + (v \dot{-} x)\!\uparrow = 0$ implies $(y \dot{-} u)\!\uparrow = 0$ and $(v \dot{-} x)\!\uparrow = 0$. $(y \dot{-} x)\!\downarrow = 0$ (and $(y \dot{-} x) \neq \langle 0, 0\rangle$) implies $x \prec y$, $(y \dot{-} u)\!\uparrow = 0$ implies $y \preceq u$ and $(v \dot{-} x)\!\uparrow = 0$ implies $v \preceq x$. By transitivity (see Theorem 1), $x \prec y$ and $y \preceq u$ imply $x \preceq u$. Moreover, $v \preceq x$ and $x \preceq u$ imply $v \preceq u$. Hence, $u \not\approx v$ (contradiction)
- $(y \dot{-} x)\!\uparrow = 0$ and $(y \dot{-} v)\!\downarrow + (u \dot{-} x)\!\downarrow = 0$. As for the previous case, Theorem 3 implies $(y \dot{-} v)\!\downarrow = 0$ and $(u \dot{-} x)\!\downarrow = 0$. $(y \dot{-} x)\!\uparrow = 0$ implies $y \prec x$, $(y \dot{-} v)\!\downarrow = 0$ implies $v \preceq y$ and $(u \dot{-} x)\!\downarrow = 0$ implies $x \preceq u$. By transitivity, $v \preceq y$ and $y \prec x$ imply $v \preceq x$. Moreover, $v \preceq x$ and $x \preceq u$ imply $v \preceq u$. Hence, $u \not\approx v$ (contradiction)

- $(y \dot{-} v)\downarrow + (u \dot{-} x)\downarrow = 0$ and $(y \dot{-} u)\uparrow + (v \dot{-} x)\uparrow = 0$. As for the previous cases, we have $v \preceq y$, $x \preceq u$, $y \preceq u$, and $v \preceq x$. $v \preceq y$ and $y \preceq u$ suffice to conclude that $u \not\approx v$.

□

Next, by resolving some $u \approx v$ in a D-Model $\mathcal{P}$ we obtain a new D-Model $\mathcal{P}[u, v]$ and there exists some poset Q that maps to $\mathcal{P}[u, v]$ under the transformation of Theorem 3.

Corollary 1. *Let $P = \langle X, \leqslant_p \rangle$ be a poset, and $\mathcal{P} = \langle X, \dot{-} \rangle$ be its corresponding D-Model. If $u \approx v$ and $\mathcal{P}[u, v] = \langle X, \dot{-}' \rangle$, then there exists a poset Q with a corresponding D-Model $\mathcal{Q}$ obtained from Theorem 3 such that $\mathcal{Q} = \mathcal{P}[u, v]$.*

Proof. Full proof in the appendix. □

This result is extremely important, as it implies that if we perform a resolution $\mathcal{Q} = \mathcal{P}[u, v]$ on the D-Model $\mathcal{P}$, then the relations of $\mathcal{P}$ will evolve according to Theorem 4. Moreover, if t is a linear extension of a D-Model $\mathcal{P}$ with $u \approx v$, then t is also a linear extension of $\mathcal{P}[u, v]$. Furthermore, let $r = \langle (u_0, v_0), \ldots, (u_n, v_n) \rangle$ be the jumps in t. Using Theorem 4, we can also show that r is a resolution sequence that resolves $\mathcal{P}$. The next lemmas formalise these results.

Lemma 2. *Let P be a poset, $\mathcal{P} = \langle X, \dot{-} \rangle$ its corresponding D-Model, t a linear extension of $\mathcal{P}$, and (u, v) a jump in t for $\mathcal{P}$. Then t is a linear extension of $\mathcal{P}[u, v]$.*

Proof. Full proof in the appendix. □

Lemma 3. *Let $P = \langle X, \leqslant_p \rangle$ be a poset, t a linear extension of P, and $r = \langle (u_0, v_0), \ldots (u_n, v_n) \rangle$ the jumps in t, ordered as in t. Then r is sound for $\mathcal{P}$.*

Proof. We prove the thesis by showing that, for all $i = 0, 1 \ldots, n$, $r_i = \langle (u_0, v_0), \ldots, (u_i, v_i) \rangle$ is sound for $\mathcal{P}$. **Base case.** $i = 0$. Clearly, $u_0 \approx v_0$. **Induction step.** Assume that for $i < n$, r_i is sound for $\mathcal{P}$. We now show that r_{i+1} is sound for $\mathcal{P}$. Let $\mathcal{P}_0 = \mathcal{P}$, $\mathcal{P}_1$, $\ldots$, $\mathcal{P}_{i+1}$ be the D-Models induced by applying r_i on $\mathcal{P}$. To prove that r_{i+1} is sound for $\mathcal{P}$, we need to prove that $u_{i+1} \approx v_{i+1}$ in $\mathcal{P}_{i+1}$. Suppose that $u_{i+1} \approx v_{i+1}$ does not hold for $\mathcal{P}_{i+1}$. Therefore, either $u_{i+1} \preceq v_{i+1}$ or $v_{i+1} \preceq u_{i+1}$ in $\mathcal{P}_{i+1}$. Suppose $u_{i+1} \preceq v_{i+1}$ (the other case is analogous). Since (u_{i+1}, v_{i+1}) was a jump for the initial D-Model $\mathcal{P}_0$, a resolution of some previous jump made $u_{i+1} \preceq v_{i+1}$. More formally, $\exists j$ with $0 \leqslant j \leqslant i$ such that $u_{i+1} \approx v_{i+1}$ in $\mathcal{P}_j$ and $u_{i+1} \preceq v_{i+1}$ in $\mathcal{P}_{j+1}$. By Corollary 1, Theorem 4 applies to all the D-Models induced by r_i as it is a sound resolution sequence. Thus, by Theorem 4, we have $u_{i+1} \preceq u_j$ and $v_j \preceq v_{i+1}$ in $\mathcal{P}_j$, and also $\mathcal{P}_i$. Recall that t has the following form: $\cdots u_j \leqslant_t v_j \cdots u_{i+1} \leqslant_t v_{i+1} \cdots$. Since $u_{i+1} \preceq u_j$ in $\mathcal{P}_i$ but $u_{i+1} \leqslant_t u_j$ does not hold, we have that t is not a linear extension of $\mathcal{P}_i$. However, by Lemma 2 (see also Lemma 6 in the appendix), t is a linear extension of $\mathcal{P}_0, \mathcal{P}_1, \ldots, \mathcal{P}_i$ (contradiction). Therefore, $u_{i+1} \approx v_{i+1}$ in $\mathcal{P}_{i+1}$, and thus r_{i+1} is sound for $\mathcal{P}$. □

We now show that if t is a linear extension for a poset P and r is the resolution sequence composed by the jumps in t for P, then r resolves $\mathcal{P}$. Then, we show that t has the minimal number of jumps iff r is optimal for $\mathcal{P}$.

Lemma 4. *Let $P = \langle X, \leqslant_p \rangle$ be a poset, t a linear extension, and r the jumps in t, ordered as in t. Then r resolves $\mathcal{P}$.*

Proof. By Lemma 3, r is sound for $\mathcal{P}$. Moreover, Lemma 2 can be used to show that t is a linear extension of $\mathcal{P}[r]$. By definition of resolution, for every jump (u, v) in r, we have $u \preceq v$ in $\mathcal{P}[r]$. Therefore, there are no jumps in t for $\mathcal{P}[r]$, and this means that all pairs of subsequent objects $x_i \leqslant_t x_{i+1}$ are comparable in $\mathcal{P}[r]$. Hence, by transitivity, there are no parities in $\mathcal{P}[r]$ (see also Lemma 7 in the appendix). □

Lemma 5. *Let P be a partial order and $\mathcal{P}$ be its corresponding D-Model. If there exists a resolution sequence $r = \langle (u_0, v_0), \ldots, (u_n, v_n) \rangle$ that resolves $\mathcal{P}$, then there exists a linear extension t with $s(P, t) \leqslant c(r)$.*

Proof. If r resolves $\mathcal{P}$, then (1) $\mathcal{P}[r]$ has a unique linear extension t and (2) t is a linear extension of $\mathcal{P}_0 = \mathcal{P}, \mathcal{P}_1, \ldots, \mathcal{P}_n$ (see also Lemma 8 in the appendix). Therefore, t is also a linear extension of P. We now show that $s(P, t) \leqslant c(r)$. Suppose that $s(P, t) > c(r)$. This implies that there is at least one jump $x \leqslant_t y$ for P such that $(x, y) \notin r$. Therefore, $x \approx y$ in $\mathcal{P}$. Suppose that for some $i = 0, \ldots, n$, we have $x \preceq u_i$ and $v_i \preceq y$ in $\mathcal{P}_i$. Then, since t is a linear extension of $\mathcal{P}_i$, we have $x \leqslant_t u_i$ and $v_i \leqslant_t y$. Since $x \leqslant_t y$ is a jump, x and y must be adjacent in t. Therefore, t is of the form:

$$\cdots \leqslant_t v_i \leqslant_t \cdots \leqslant_t x \leqslant_t y \leqslant_t \cdots \leqslant_t u_i \leqslant_t \cdots$$

But $u_i \prec v_i$ in $\mathcal{P}_{i+1}$ by Theorem 2. Since $u_i \not\leqslant_t v_i$, t is not a linear extension of $\mathcal{P}_{i+1}$ (contradiction). So, $\forall i = 0, \ldots, n$ we have that $x \preceq u_i$ and $v_i \preceq y$ does not hold in $\mathcal{P}_i$. Therefore, since $x \approx y$ in $\mathcal{P}_0$, by Theorem 4 we also have that $x \approx y$ in $\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_{n+1} = \mathcal{P}[r]$ (contradiction). □

Corollary 2. *Let P be a partial order and $\mathcal{P}$ its corresponding D-Model. Then $s(P) = c(\mathcal{P})$.*

Proof. Suppose $s(P) < c(\mathcal{P})$. Then, there exists a linear extension t of P with $s(P, t) < c(\mathcal{P})$ jumps. By Lemma 4, the resolution sequence r composed of the jumps in t resolves $\mathcal{P}$ and $c(r) = s(P, t)$. Hence, $c(r) < c(\mathcal{P})$ (contradiction). Suppose $c(\mathcal{P}) < s(P)$. Let r be the resolution sequence such that $c(r) = c(\mathcal{P})$. Theorem 5 implies that there exists a linear extension t of P with $s(P, t) \leqslant c(r) = c(\mathcal{P})$. Since $c(\mathcal{P}) < s(P)$, we have $s(P, t) < s(P)$ (contradiction). □

Theorem 5. *The Resolution Number Problem is NP-complete.*

Proof. We first show that the problem is NP. Recall that a resolution cannot "introduce" new parities in the resulting model. Therefore, the maximum number of pairs requiring resolutions in a D-Model M is equal to the number of pairs $(x, y) \in X \times X$ such that $x \approx y$ in M. Therefore, a non-deterministic machine can guess a sequence of at most k

resolutions $r = \langle (v_0, u_0), \ldots, (v_k, u_k) \rangle$, and then we can check in polynomial time if r is sound and resolves M. To show hardness, we polynomially reduce the jump number problem to the resolution number problem. Let $P = \langle X, \leqslant \rangle$ be any poset. We can obtain $\mathcal{P}$ using Theorem 3 in $\mathcal{O}(|X|^2)$ time. Moreover, by Corollary 2, we have that $s(P) = c(\mathcal{P})$. Hence, for an integer k, we have $s(P) \leqslant k$ iff $c(\mathcal{P}) \leqslant k$. $\square$

Searching for Optimal Resolution Sequences

In this section, we focus on the problem of finding an optimal resolution sequence for a difference model M. To do so, we propose an encoding to numeric planning. Intuitively, given a difference model M, we create a new planning problem Π_M whose optimal solutions correspond to the optimal resolution sequences for M. As we will see, the numeric planning formalism provides a simple and elegant way of encoding D-Models. We start with some brief background on numeric planning with conditional effects (Fox and Long 2003; Bonassi et al. 2025).

A numeric planning problem Π is a tuple $\langle V, I, G, A, c \rangle$, where V is a set of numeric variables, and the state space of P is the set of all valuations for V. Numeric planning also allows Boolean state variables, but we omit them as not required for our encoding. Let $v \in V$, and s be a state, $s(v)$ that indicates the value of v in s. I is the *initial* state. Let L be the language of propositional formulas that have numeric conditions as terminals, i.e., $\xi\{\geqslant, >, =\}0$ with ξ a numeric expression over V and $\mathbb{Q}$. For a numeric expression ξ, we write $s(\xi)$ to indicate the value of ξ in s. G, the goal, is an element of L. A is a set of actions such that each $a \in A$ is a pair $\langle \mathrm{pre}(a), \mathrm{eff}(a) \rangle$ where $\mathrm{pre}(a) \in L$ and $\mathrm{eff}(a)$ is a set of *conditional effects* describing the action effects. A conditional effect is a pair $\phi \rhd v := \xi$ where $\phi \in L$, $v \in V$, and ξ is an arithmetic expression. Finally, c is a cost function assigning a positive value to each action a.

An action a is applicable in s if $s \models \mathrm{pre}(a)$, and its execution produces a new state $s' = s[a]$ defined accoring to the action's effects. Intuitively, a conditional effect $\phi \rhd v := \xi$ indicates that when the action is executed the variable v gets assigned to $s(\xi)$ in s' only if s satisfies ϕ. Formally, for every $v \in V$, $s'(v)$ is defined as follows:

$$s'(v) = \begin{cases} s(\xi) & \text{if } \exists\, \phi \rhd v := \xi \in \mathrm{eff}(a) \text{ such that } s \models \phi \\ s(v) & \text{otherwise} \end{cases}$$

We assume that the problem has no conflicting effects. A plan for Π is a sequence of actions $\pi = \langle a_0, \ldots, a_n \rangle$ from A. The execution of π from I induces a sequence of states $\langle s_0 = I, s_1 = s_0[a_0], \ldots, s_{n+1} = s_n[a_n]] \rangle$. We say that π is valid for Π if for each $i \in \{1, \ldots, n\}$, $s_i \models \mathrm{pre}(a_i)$, and $s_{n+1} \models G$. The cost of a valid plan π is $c(\pi) = \sum_{a \in \pi} c(a)$, and a plan is optimal if it has minimal cost.

We now present our encoding. Given an initial difference model $M_0 = \langle X, \dot{-} \rangle$ we define a numeric planning problem $\Pi_{M_0} = \langle V, I, G, A, c \rangle$ as follows.

Variables. Numeric variables encode the value of all approximate differences. Specifically, for each $y, x \in X$, we have two variables: $ub_{y,x}$ and $lb_{y,x}$ for the upper and lower bounds of $(y \dot{-} x)$. Formally, $V = \bigcup_{y,x \in X} \{lb_{y,x}, ub_{y,x}\}$.

Initial State. The initial state captures the value of $(y \dot{-} x)$ in M_0 for each $y, x \in X$. Formally, $I = \bigcup_{y,x \in X} \{lb_{y,x} := (y \dot{-} x)\downarrow, ub_{y,x} := (y \dot{-} x)\uparrow\}$

Goal. To find the optimal resolution, we need to transform M_0 into a new D-Model without parities. Recall that $y \approx x$ iff $(y \dot{-} x)\downarrow < 0$ and $(y \dot{-} x)\uparrow > 0$. Hence, we can enforce a goal state to have no parities by setting $G = \bigwedge_{y,x \in X} \neg(lb_{y,x} < 0 \wedge ub_{y,x} > 0)$

Actions. Each action of the problem encodes a possible resolution. Therefore, the set of actions is defined as $A = \bigcup_{u,v \in X} \{resolve_{u,v}\}$. To ensure that plans encode sound resolution sequences, we must ensure that an action $resolve_{u,v}$ is executed only when $u \approx v$. Therefore, we set $\mathrm{pre}(resolve_{u,v}) = (lb_{u,v} < 0 \wedge ub_{u,v} > 0)$. Moreover, we set the effect of each action to perform the resolution as in Definition 2. For some $u \approx v$ we wish to resolve, let $\overline{r}_{y,x}$ denote $lb_{y,v} + lb_{u,x}$ and $\underline{s}_{y,x}$ denote $ub_{y,u} + ub_{v,x}$ for some $y, x \in X$. To perform the resolution $M[u, v]$ for some model M, we need to set $lb_{y,x}$ to $\max(lb_{y,x}, \overline{r}_{y,x})$ and $ub_{y,x}$ to $\min(ub_{y,x}, \underline{s}_{y,x})$ for every $y, x \in X$. We can do so with two conditional effects:

$$(lb_{y,x} < \overline{r}_{y,x}) \rhd lb_{y,x} := \overline{r}_{y,x} \qquad (\mathrm{eff}^{lb}_{y,x})$$

$$(ub_{y,x} > \underline{s}_{y,x}) \rhd ub_{y,x} := \underline{s}_{y,x} \qquad (\mathrm{eff}^{ub}_{y,x})$$

Hence, we have $\mathrm{eff}(resolve_{u,v}) = \bigcup_{y,x \in X} \{\mathrm{eff}^{lb}_{y,x}, \mathrm{eff}^{ub}_{y,x}\}$. Finally, we have $c(a) = 1$ for every action $a \in A$.

We now briefly analyze the correctness of the encoding. Let M_0 be a D-Model and Π_{M_0} its corresponding planning problem. For a state s of Π_{M_0}, we denote by $s|_M$ the D-Model encoded within s, that is, $s|_M = \langle X, \dot{-} \rangle$ where $(y \dot{-} x) = \langle s(lb_{y,x}), s(ub_{y,x}) \rangle$ for every $y, x \in X$. It can be proven that every reachable state s of Π_{M_0} is such that $s|_M$ is a D-Model. This is because (i) by definition $I|_M = M_0$, (ii) every action $resolve_{u,v}$ can be executed in a state s only if $u \approx v$ in $s|_M$, and (iii) actions perform the resolution as in Definition 2, that is, executing $resolve_{u,v}$ in a state s induces a new state s' such that $s'|_M = s|_M[u, v]$. Therefore, every valid plan $\pi = \langle resolve_{u_0,v_0}, \ldots, resolve_{u_n,v_n} \rangle$ of Π_{M_0} corresponds to a resolution sequence $r = \langle (u_0, v_0), \ldots, (u_n, v_n) \rangle$ that solves M_0. Since the cost of every action is unitary, π is optimal iff r is optimal. For these reasons, the encoding is correct.

Theorem 6. *Let M_0 be a D-Model and Π_{M_0} its corresponding numeric planning problem. Any valid plan π for Π_{M_0} can be mapped to a resolution sequence r that solves M_0, and vice versa. Moreover, π is optimal iff r is optimal.*

Experiments

Currently, no automatic approach for finding a minimal resolution sequence for D-Models exists. Therefore, in our experimental analysis, we focus on evaluating the effectiveness and scalability of current state-of-the-art numeric planners in finding an optimal resolution sequence for a set of benchmark D-Models. To do so, we created 20 D-Models with increasing numbers of items and parities by randomly sampling approximate differences within the interval $[-100, 100]$, and expanding these limits if necessary to maintain triangular inequality. We then encoded these D-Models using the previously presented approach. We denote these instances by Π^{CE}. Currently, numeric planners either lack support for conditional effects or do not incorporate them into optimal planning heuristics. To broaden the range of available planners, we created another set of equivalent instances Π^{COMP} by compiling conditional effects away. Our compilation is specific to our domain and, similar to existing polynomial compilations (Nebel 2000; Bonassi et al. 2025), replaces conditional effects with sequences of actions yielding the same successor states. To ensure that the cost of solution plans corresponds to the number of resolutions required to resolve a D-Model, the cost of the additional actions introduced to handle conditional effects is set to zero. All instances are provided in the supplementary material.

For optimal planners, we considered an A^* search guided by the blind heuristic h^{blind} and the h^{max}_{hbd} heuristic (Scala et al. 2020), both implemented in ENHSP (Scala et al. 2016). Moreover, we considered NLM-CutPlan (NLM) (Kuroiwa, Shleyfman, and Beck 2023) and OMTplan (Leofante 2023) (OMT), two optimal planners that participated in the numeric track of the 2023 International Planning Competition (Taitler et al. 2024). NLM is based on Numeric Fast Downward (Aldinger and Nebel 2017), and performs an A^* search guided by a variant of the numeric LM-Cut heuristic (Kuroiwa et al. 2022) that exploits the bounds of numeric variables. Differently, OMT is an optimal planner based on Satisfiability Modulo Theories (Barrett et al. 2021). We run h^{blind} on Π^{CE} instances, as it is the only configuration supporting conditional effects, while we run all configurations over Π^{COMP}. We collect the runtime, solution cost, and number of nodes expanded by the search. Experiments were run on a 24-Core AMD Ryzen Threadripper 7960X processor with runtime and memory limits of 60 minutes and 16 GB.

Table 2 reports the optimal solution cost and runtime achieved by all systems. Each instance is denoted by X–Y, where X is the number of items and Y is the number of parities in the D-Model. For example, instance 5–8 encodes a D-Model with 5 items and 8 parities. The table reports results only for the first 12 benchmark instances, as all planners ran out of time or memory on the larger problems. The full table is provided in the appendix. We observe that the best results are obtained by h^{blind} on the Π^{CE} problems, whereas performance degrades on Π^{COMP}. Intuitively, the conditional effect compilation polynomially increases the size of solution plans (Nebel 2000), making the search more difficult. Indeed, the same h^{blind} configuration solves 7 Π^{COMP} instances, compared to 12 for Π^{CE}. On average, h^{blind} on Π^{CE} expands 14168 nodes versus 1286305 nodes when run on Π^{COMP}. Moreover, while the length of optimal plans for Π^{CE} matches the cost of the optimal solution, the plans produced by h^{blind} in Π^{COMP} contain, on average, 78.71 actions.

We also observe that current planners are less effective on Π^{COMP} than a blind search on Π^{CE}. Specifically, we observe that h^{max}_{hbd} does not provide effective heuristic guidance on Π^{COMP} instances: on average, the number of nodes expanded by h^{max}_{hbd} (1243978) is similar to that of h^{blind} (1286305). This is because h^{max}_{hbd} was designed for problems where variables can only be increased or decreased by a constant, whereas Π^{COMP} introduces variable assignments and linear expressions on the right-hand side of effects. Interestingly, in one Π^{COMP} instance (10-34) h^{blind} runs out of memory, while h^{max}_{hbd} manages to find a solution. In principle, NLM and OMT should be better suited for this class of problem, yet they exhibit different weaknesses. NLM frequently exceeds memory limits before the start of the search. OMT, on the other hand, operates similarly to planning-as-SAT approaches (Rintanen 2021), incrementally constructing a formula ϕ_n that is satisfiable iff a plan of length n achieves the goal. This makes OMT less effective in Π^{COMP}, where solution plans are typically long due to conditional effect compilation. For example, the only instance solved by OMT features 24 actions while only two resolutions are required to solve the corresponding D-Model.

		Π^{CE}	Π^{COMP}			
Problem	Cost	h^{blind}	h^{blind}	h^{max}_{hbd}	NLM	OMT
5-8	2	0.48	**0.25**	0.52	11.51	313.02
5-9	3	0.59	**0.35**	0.64	95.50	TO
6-12	4	1.10	**1.08**	7.17	TO	TO
6-13	4	**1.11**	1.24	9.79	TO	TO
7-20	5	**9.99**	87.38	955.89	TO	TO
7-21	6	**72.08**	MO	MO	TO	TO
8-23	4	**2.96**	18.28	246.93	MO	TO
8-24	4	**5.43**	25.81	358.91	MO	TO
9-28	4	**50.97**	MO	MO	MO	TO
9-29	5	**86.77**	MO	MO	MO	TO
10-34	4	**29.24**	MO	3172.05	MO	TO
10-37	4	**180.71**	MO	TO	MO	TO

Table 2: Solution cost and runtime (in seconds) achieved by all systems. "TO" indicates the planner timing out while "MO" indicates the planner running out of memory.

Conclusions

This work adopts Difference Models as a framework for representing and reasoning about parity. We formalised the Resolution Number Problem, which asks for the minimal number of resolutions to eliminate all parities, and established its computational complexity by proving NP-completeness via a reduction from the Jump Number Problem for posets. Furthermore, we proposed a practical approach for computing optimal resolution sequences by encoding the problem into numeric planning and using established planning techniques. Our results show that blind search is a viable method for solving the considered class of problems, whereas current state-of-the-art optimal planners struggle to scale due to their lack of support for conditional effects. Future work will look to extend optimal planners for conditional effects and develop domain specific heuristics. Moreover, we plan to integrate planning algorithms for Difference Models with interactive decision-support in applications where parity-aware reasoning is essential for system trustworthiness.

Acknowledgments

This work was supported by the joint UKRI and AISI-DSIT Systemic Safety Grant [grant number UKRI854].

References

Abeywickrama, D. B.; Bennaceur, A.; Chance, G.; Demiris, Y.; Kordoni, A.; Levine, M.; Moffat, L.; Moreau, L.; Mousavi, M. R.; Nuseibeh, B.; et al. 2023. On specifying for trustworthiness. *Communications of the ACM*, 67(1): 98–109.

Aldinger, J.; and Nebel, B. 2017. Interval Based Relaxation Heuristics for Numeric Planning with Action Costs. In *KI*, volume 10505 of *Lecture Notes in Computer Science*, 15–28. Springer.

Barrett, C. W.; Sebastiani, R.; Seshia, S. A.; and Tinelli, C. 2021. Satisfiability Modulo Theories. In *Handbook of Satisfiability*, volume 336 of *Frontiers in Artificial Intelligence and Applications*, 1267–1329. IOS Press.

Bender, E. M.; Gebru, T.; McMillan-Major, A.; and Shmitchell, S. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, 610–623.

Bogart, K. P.; and Trenk, A. N. 1994. Bipartite tolerance orders. *Discrete Mathematics*, 132(1): 11–22.

Bonassi, L.; Espasa, J.; Percassi, F.; and Scala, E. 2025. Conditional Effects in Numeric Planning Reloaded. In *28th European Conference on Artificial Intelligence*. IOS Press.

Bouchitté, V.; and Habib, M. 1987. NP-completeness properties about linear extensions. *Order*, 4(2): 143–154.

Chakraborti, T.; Sreedharan, S.; and Kambhampati, S. 2018. Explicability versus Explanations in Human-Aware Planning. In *AAMAS*, 2180–2182. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM.

Chang, R. 2002. The Possibility of Parity. *Ethics*, 112(4): 659–688.

Chang, R. 2017. Hard choices. *Journal of the American Philosophical Association*, 3(1): 1–21.

Colbourn, C. J.; and Pulleyblank, W. R. 1985. Minimizing setups in ordered sets of fixed width. *Order*, 1(3): 225–229.

Davey, B. A.; and Priestley, H. A. 2002. *Introduction to Lattices and Order, Second Edition*. Cambridge University Press.

Dobbe, R.; Gilbert, T. K.; and Mintz, Y. 2021. Hard choices in artificial intelligence. *Artif. Intell.*, 300: 103555.

Fine, K. 2026. Modelling Parity and Imprecision. To appear in *Oxford Studies in Epistemology*.

Fishburn, P. C. 1970. Intransitive indifference with unequal indifference intervals. *Journal of Mathematical Psychology*, 7(1): 144–149.

Fox, M.; and Long, D. 2003. PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. *J. Artif. Intell. Res.*, 20: 61–124.

Holbrook, C.; Holman, D.; Clingo, J.; and Wagner, A. R. 2024. Overtrust in ai recommendations about whether or not to kill: Evidence from two human-robot interaction studies. *Scientific reports*, 14(1): 19751.

Kierstead, H. A. 1986. NP-completeness results concerning greedy and super greedy linear extensions. *Order*, 3(2): 123–134.

Kuroiwa, R.; Shleyfman, A.; and Beck, J. C. 2023. Extracting and Exploiting Bounds of Numeric Variables for Optimal Linear Numeric Planning. In *ECAI*, volume 372 of *Frontiers in Artificial Intelligence and Applications*, 1332–1339. IOS Press.

Kuroiwa, R.; Shleyfman, A.; Piacentini, C.; Castro, M. P.; and Beck, J. C. 2022. The LM-Cut Heuristic Family for Optimal Numeric Planning with Simple Conditions. *J. Artif. Intell. Res.*, 75: 1477–1548.

Leofante, F. 2023. OMTPlan: A Tool for Optimal Planning Modulo Theories. *J. Satisf. Boolean Model. Comput.*, 14(1): 17–23.

Luce, R. D. 1956. Semiorders and a Theory of Utility Discrimination. *Econometrica*, 24(2): 178–191.

McElfresh, D. C.; Chan, L.; Doyle, K.; Sinnott-Armstrong, W.; Conitzer, V.; Borg, J. S.; and Dickerson, J. P. 2021. Indecision Modeling. In *AAAI*, 5975–5983. AAAI Press.

Nebel, B. 2000. On the Compilability and Expressive Power of Propositional Planning Formalisms. *J. Artif. Intell. Res.*, 12: 271–315.

Nguyen, D.; Bhattacharyya, R.; Colombatto, C.; Fleming, S.; Posner, I.; and Hawes, N. 2025. Group Decision-Making in Robot Teleoperation: Two Heads are Better Than One. In *HRI*, 489–500. IEEE.

Noothigattu, R.; Gaikwad, S. N. S.; Awad, E.; Dsouza, S.; Rahwan, I.; Ravikumar, P.; and Procaccia, A. D. 2018. A Voting-Based System for Ethical Decision Making. In *AAAI*, 1587–1594. AAAI Press.

Rintanen, J. 2021. Planning and SAT. In *Handbook of Satisfiability*, volume 336 of *Frontiers in Artificial Intelligence and Applications*, 765–789. IOS Press.

Scala, E.; Haslum, P.; Thiébaux, S.; and Ramírez, M. 2016. Interval-Based Relaxation for General Numeric Planning. In *ECAI*, volume 285 of *Frontiers in Artificial Intelligence and Applications*, 655–663. IOS Press.

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.

Stephens, A.; Budd, M.; Staniaszek, M.; Casseau, B.; Duckworth, P.; Fallon, M.; Hawes, N.; and Lacerda, B. 2024. Planning under uncertainty for safe robot exploration using Gaussian process prediction. *Autonomous Robots*, 48(7): 18.

Szpilrajn, E. 1930. Sur l'extension de l'ordre partiel. *Fundamenta mathematicae*, 16(1): 386–389.

Taitler, A.; Alford, R.; Espasa, J.; Behnke, G.; Fiser, D.; Gimelfarb, M.; Pommerening, F.; Sanner, S.; Scala, E.; Schreiber, D.; Segovia-Aguas, J.; and Seipp, J. 2024. The 2023 International Planning Competition. *AI Mag.*, 45(2): 280–296.

Wilson, J.; and Hauert, S. 2022. Search space illumination of robot swarm parameters for trustworthy interaction. In *International Symposium on Distributed Autonomous Robotic Systems*, 173–186. Springer.

Yun, H. S.; and Bickmore, T. 2025. Online health information–seeking in the era of large language models: cross-sectional web-based survey study. *Journal of Medical Internet Research*, 27: e68560.

Zahedi, Z.; Sengupta, S.; and Kambhampati, S. 2024. 'Why Didn't You Allocate This Task to Them?' Negotiation-Aware Task Allocation and Contrastive Explanation Generation. In *AAAI*, 10243–10251. AAAI Press.

Zahedi, Z.; Verma, M.; Sreedharan, S.; and Kambhampati, S. 2023. Trust-Aware Planning: Modeling Trust Evolution in Iterated Human-Robot Interaction. In *HRI*, 281–289. ACM.