

# Planning as Goal Recognition: Deriving Heuristics from Intention Models

Giacomo Rosa<sup>1,\*</sup>, Jean Honorio<sup>1,\*</sup>, Nir Lipovetzky<sup>1,\*</sup>, Sebastian Sardiña<sup>2,§</sup>

<sup>1</sup>The University of Melbourne, Australia

<sup>2</sup>RMIT University, Melbourne, Australia

\*{g.rosa, jean.honorio, nir.lipovetzky}@unimelb.edu.au, §sebastian.sardina@rmit.edu.au

## Abstract

Classical planning aims to find a sequence of actions, a plan, that maps a starting state into one of the goal states. If a trajectory appears to be leading to the goal, should we prioritise exploring it? Seminal work in goal recognition (GR) has defined GR in terms of a classical planning problem, adopting classical solvers and heuristics to recognise plans. We come full circle, and study the adoption and properties of GR-derived heuristics for seeking solutions to classical planning problems. We propose a new divergence-based framework for assessing goal intention, which informs a new class of efficiently-computable heuristics. As a proof of concept, we derive two such heuristics, and show that they can already yield improvements for top-scoring classical planners. Our work provides foundational knowledge for understanding and deriving probabilistic intention-based heuristics for planning.

**Extended version** — <https://arxiv.org/abs/2603.14824>

## Introduction

We study the connection between goal recognition (GR) and classical planning by characterizing planning heuristics as mechanisms for assessing their goal-intentionality, thus doing a full loop from Ramirez and Geffner (2009, 2010)’s seminal work on GR as planning. Classical planning is the field of AI that seeks to find a sequence of actions, a *plan*, that maps an initial state in a problem into a state that satisfies a specific goal condition. In classical planning, actions are deterministic, states are fully observable and represented through binary variables (*facts*), and no other actions occur outside the plan (i.e., static environment). A common strategy adopted by solvers is to perform a search over the state space, using heuristics which estimate the distance to the goal, such as the FF (Bonet and Geffner 2001; Hoffmann and Nebel 2001) and Landmark (Porteous, Sebastia, and Hoffmann 2001) heuristics, to guide the search. Other techniques have also proven to be effective in enhancing search efficiency. These include identifying *helpful actions* (Hoffmann and Nebel 2001), which prioritise operators likely to contribute to goal achievement, leveraging *novelty* measures (Lipovetzky and Geffner 2012, 2017), which favor exploration of states exhibiting previously unseen combinations of features, and *dominance pruning* (Torralba and

Hoffmann 2015), which prunes states that provably cannot yield better solutions than previously explored states. These techniques do not estimate goal distance directly; rather, they prioritize states or nodes that aid the search through alternative mechanisms.

The goal recognition task involves an observer inferring an acting agent’s goals or plans based on partial observations of its behaviour (Sukthankar et al. 2014).<sup>1</sup> Traditional approaches rely on a predefined *plan library*, which encodes known plans for implicit goals, allowing recognition through matching observed actions to entries in the library (Kautz and Allen 1986). The *goal recognition as planning* framework (Ramirez and Geffner 2009, 2010) casts recognition as a planning problem itself: a (declarative) goal is considered more likely if the observed actions align with an optimal or near-optimal plan to achieve it. Recent contributions have extended this paradigm to account for irrational behaviour of agents (Masters and Sardiña 2021), as well as adopting estimated measures and using information contained in the *effects* of observed actions to recognise goals and plans (Pereira, Oren, and Meneguzzi 2017; Wilken et al. 2024).

In light of these recent developments, we revisit search for resource-bounded agents that seek to perform an intelligent exploration. Due to resource limitations, such agents bias the search towards some fragment of all possible traces (Pollack 1992; Bratman, Israel, and Pollack 1988). We analyze the intentionality of these traces with the lens of work in GR, under the intuition that some traces are seen as observations that are “more intended” towards the goal than others, and push our search algorithms to explore more intended paths. This process is framed as a GR problem where, rather than estimating the intentionality of one observation towards multiple goals, we aim to assign and compare the intentionality of different observations towards a distinguished goal.

**Contributions.** Our primary objective is to establish a novel conceptual framework that views heuristics as judges of the intentionality of discovered trajectories in a planning problem. We first present a plan-library model of GR for a resource-bounded agent in a planning domain. This allows us to define the goal-intentionality of observed traces, and

<sup>1</sup>Other common terms are Plan Recognition (PR) or Intention Recognition (IR). While subtle differences exist among them, in this paper, we shall use these terms interchangeably.

study the properties of solvers that use this quantity to seek plans. A core contribution that separates our work from prior GR approaches is a new formulation of intentionality in which the likelihood is derived from the divergence between information in the observation and that in a description of goal-intended trajectories. Our plan-library model and proposed heuristics are both shown to belong to this family of estimators. Our theory then informs a new class of *intention-based* heuristics for classical planning. As a proof of concept, we present two such heuristics, which we show help improve the state of the art in classical planning benchmarks. We tie our results to our framework, providing experimental evidence of theorised properties of our search paradigm.

## Preliminaries

The classical planning model is defined as  $\Phi = \langle S, s_0, S_G, A, f \rangle$ , where  $S$  is the discrete finite state space,  $s_0 \in S$  is the initial state,  $S_G \subseteq S$  is the set of goal states,  $A$  is the set of (deterministic) actions, and  $f : A \times S \mapsto S$  denotes the model partial transition function, with  $f(a, s)$  denoting the next state  $s' \in S$  after applying action  $a \in A$  in state  $s \in S$ . When  $f$  is undefined, the action is not applicable in the state. We write  $A(s)$  to denote the set of actions applicable in state  $s$ , i.e.,  $A(s) = \{a \in A \mid f(a, s) \text{ is defined}\}$ . A solution to a classical planning model is given by a **plan**, a sequence of actions  $\langle a_0, \dots, a_m \rangle$  that induces a state sequence  $\langle s_0, \dots, s_{m+1} \rangle$  such that  $a_i \in A(s_i)$ ,  $s_{i+1} = f(a_i, s_i)$ , and  $s_{m+1} \in S_G$  for  $i \in \{0, \dots, m\}$ .

A STRIPS<sup>2</sup> (Fikes and Nilsson 1971) problem is defined through tuple  $\mathcal{P} = \langle F, A, I, G \rangle$ , where  $F$  denotes the set of boolean variables, or fluents,  $A$  is the set of actions,  $I \subseteq F$  is the set of atoms that fully describe the initial state, and  $G \subseteq F$  is the partial assignment that describes goal states. We assume unit cost actions in this work.

**Planning Model Notation.** Given a Classical Planning problem  $\mathcal{P}$ , a **trajectory** denotes a sequence of alternating states and actions  $\langle s_k, a_k, s_{k+1}, a_{k+1}, \dots, s_m, a_m, s_{m+1} \rangle$ , where both the first and last elements are states, such that  $s_i \in S$ ,  $a_i \in A(s_i)$ , and  $s_{i+1} = f(s_i, a_i)$ ; where  $k \leq i < m + 1$ . Every trajectory induces two projections: an **a-trajectory**, which is the sequence of actions  $\langle a_k, a_{k+1}, \dots, a_m \rangle$ , and an **s-trajectory**, which is the sequence of states  $\langle s_k, s_{k+1}, \dots, s_{m+1} \rangle$ . We use  $\mathcal{T}(s_i, \pi)$  to represent the s-trajectory induced by an a-trajectory  $\pi$  applied from state  $s_i$ . We use  $\mathcal{L}(s, \pi)$  to denote the last state  $s_{m+1} \in S$  in the s-trajectory  $\mathcal{T}(s, \pi)$ . For simplicity, when  $s$  is  $s_0$  (the initial state of  $\mathcal{P}$  as per  $I$ ), we just write  $\mathcal{T}(\pi)$  and  $\mathcal{L}(\pi)$ , resp.

We place two constraints on considered trajectories: 1) **acyclic**: no state may appear more than once in a trajectory; and 2) **non-goal-extending**: goal states can only appear as the last state of the trajectory. These are reasonable assumptions, as any cycle is redundant and extending a-trajectories beyond a goal state is superfluous for finding plans towards a single goal. Given  $S_i, S_j \subseteq S$ , we use  $\Pi(S_i, S_j)$  to denote the set of *acyclic* and *non-goal-extending* a-trajectories

that can be applied to a state  $s_i \in S_i$  to yield a valid s-trajectory that begins at state  $s_i$  and ends at state  $s_j \in S_j$ . A **plan** is then an a-trajectory  $\pi \in \Pi(\{s_0\}, S_G)$ ; in other words, an a-trajectory that, when applied to  $s_0$ , reaches a valid goal state  $s_g \in S_G$ . Note that more than one sequence of actions (a-trajectories) may yield the same history of states (s-trajectories). Similarly, we use **I-reachable a-trajectory** to refer to all a-trajectories  $\pi \in \Pi(\{s_0\}, S)$ . We adopt the definition of an **observation sequence** from previous plan and goal recognition literature (Ramirez and Geffner 2009; Masters and Sardiña 2021) as *any* sequence of actions  $\langle o_1, \dots, o_m \rangle$ , with  $o_i \in A$ . An action sequence **satisfies** an observation sequence iff it **embeds** it, meaning that there is a monotonic function  $g$  that maps each observation  $o_i \in A$  to the index of an identical action in the action sequence such that  $g(o_i) < g(o_j)$  for all  $j > i$ . It follows from the above definitions that, given the set  $\mathcal{O}$  of all possible observation sequences for problem  $\mathcal{P}$ ,  $\Pi(\{s_0\}, S_G) \subseteq \Pi(\{s_0\}, S) \subseteq \Pi(S, S) \subseteq \mathcal{O}$ .

An a-trajectory  $\pi'$  **contains** a-trajectory  $\pi$ , written  $\pi \sqsubseteq \pi'$ , iff there exist, possibly empty, sequences of actions  $\alpha$  and  $\beta$  such that  $\pi' = \alpha \cdot \pi \cdot \beta$ . This relation is reflexive, i.e., an a-trajectory contains itself. An a-trajectory  $\pi'$  **extends** an a-trajectory  $\pi$ , written  $\pi \sqsubseteq_{\text{pfx}} \pi'$ , iff there exists a, possibly empty, suffix  $\beta$  such that  $\pi' = \pi \cdot \beta$ . We define the set of **maximal a-trajectories**  $\mathcal{M}$  as the set of I-reachable a-trajectories that are not extended by any other I-reachable a-trajectory in  $\Pi(\{s_0\}, S)$ :  $\mathcal{M} = \{\pi \in \Pi(\{s_0\}, S) \mid \forall \pi' \in (\Pi(\{s_0\}, S) \setminus \{\pi\}), \pi \not\sqsubseteq_{\text{pfx}} \pi'\}$ . Thus,  $\Pi(\{s_0\}, S_G) \subseteq \mathcal{M} \subseteq \Pi(\{s_0\}, S)$ . We also define the operator  $\alpha_{\text{rank}}$ , which indicates that two quantities induce the same ranking (the ordering is preserved):  $k \alpha_{\text{rank}} l := k(x) < k(y) \iff l(x) < l(y)$ .

## Intention-Based Search

Goal recognition problems assume two agents: an acting agent, which follows a (hidden) trajectory, and an observer agent, whose task is to infer the intention of the acting agent given an observation, a partial trace of the actor's full trajectory. We adapt this idea to the search problem in planning by imagining a one-vs-all GR problem, where the actor is either directed or not directed towards a single goal, and with known initial state. The observer must determine whether the actor is following a plan that satisfies the problem's goal given a partial trace, and can therefore be thought of as a heuristic that determines the goal intention of an observation. Given multiple observations in an open list, we can then rank them based on their assigned probability of being intended towards the goal, preferring higher ranked observations for expansion as a means of guiding the search. We begin by assuming that the observer may be resource bounded, and as such may not have knowledge of all maximal a-trajectories starting from  $s_0$ . Rather, it has prior knowledge of a non-empty subset  $\hat{\mathcal{M}} \subseteq \mathcal{M}$  of sampled candidate maximal a-trajectories, which it uses to infer the actor's goal-intention. The observer is also subject to beliefs regarding the behaviour of the acting agent, which are expressed by assigning a weight to every maximal a-trajectory, a measure

<sup>2</sup>Stanford Research Institute Problem Solver

of preference for that a-trajectory. For example, if the observer believes that the actor is rational, it will assume that it is more likely to follow optimal or near-optimal paths towards its objective, and as such assign greater weights to shorter maximal a-trajectories. Alternatively, it may assign uniform weights if its belief is that the actor prioritizes all maximal a-trajectories equally.

### IRPL Model

We provide the *I-Reachable Plan-Library* (IRPL) model, that only considers I-reachable a-trajectories in a planning problem as valid observation sequences, and derives probabilities relative to an implicit library of sampled maximal a-trajectories, and the subset of those a-trajectories that constitute plans. This allows us to illustrate the usefulness of adopting such probabilities as heuristic signals in a planning problem, under the simplified scenario where probabilistic events are explicitly observable. Given problem description  $\langle F, A, I, G \rangle$  and the set of all I-reachable a-trajectories  $\Pi(\{s_0\}, S)$ , the observer samples a set  $\hat{\mathcal{M}}$  of known maximal a-trajectories starting at the initial state  $s_0$ , and assigns a weight to each maximal a-trajectory according to a *weight function*  $w : \mathcal{M} \rightarrow \mathbb{R}^+$ . Let the set of sampled plans be  $\hat{\mathcal{M}}_G = \Pi(\{s_0\}, S_G) \cap \hat{\mathcal{M}}$ . For I-reachable a-trajectories  $O \in \Pi(\{s_0\}, S)$ , let  $C(O) = \{\pi' \in \hat{\mathcal{M}} \mid O \sqsubseteq_{\text{pfx}} \pi'\}$  be the set of all sampled maximal a-trajectories that extend  $O$ . Let  $C_G(O) = \{\pi' \in \hat{\mathcal{M}}_G \mid O \sqsubseteq_{\text{pfx}} \pi'\}$  be the set of sampled plans that extend  $O$ , and  $C_{-G}(O) = \{\pi' \in (\hat{\mathcal{M}} \setminus \hat{\mathcal{M}}_G) \mid O \sqsubseteq_{\text{pfx}} \pi'\}$  be the sampled maximal non-plans extending  $O$ , such that  $C_G(O) \cup C_{-G}(O) = C(O)$  and  $C_G(O) \cap C_{-G}(O) = \emptyset$ .

We define sets of maximal a-trajectories  $\mathcal{E} \subseteq \hat{\mathcal{M}}$  that imply underlying events of interest, with probability

$$P(\mathcal{E}) = \sum_{\pi' \in \mathcal{E}} w(\pi') / \sum_{\pi'' \in \hat{\mathcal{M}}} w(\pi'').$$

Thus,  $P(G)$  is the event that  $\pi \in \hat{\mathcal{M}}$  is a plan:

$$P(G) := P(\hat{\mathcal{M}}_G) = \sum_{\pi' \in \hat{\mathcal{M}}_G} w(\pi') / \sum_{\pi'' \in \hat{\mathcal{M}}} w(\pi''). \quad (1)$$

Similarly,  $P(\neg G) := P(\hat{\mathcal{M}} \setminus \hat{\mathcal{M}}_G)$ .  $P(O)$  is the event that  $\pi \in \hat{\mathcal{M}}$  extends  $O$ :

$$P(O) := P(C(O)) = \sum_{\pi' \in C(O)} w(\pi') / \sum_{\pi'' \in \hat{\mathcal{M}}} w(\pi''). \quad (2)$$

We can then obtain conditional probability  $P(O \mid G) := P(C(O) \mid \hat{\mathcal{M}}_G)$ :

$$P(O \mid G) = \sum_{\pi' \in C_G(O)} w(\pi') / \sum_{\pi'' \in \hat{\mathcal{M}}_G} w(\pi''), \quad (3)$$

where  $C_G(O) = C(O) \cap \hat{\mathcal{M}}_G$ .<sup>3</sup> Finally, Bayesian posterior  $P(G \mid O)$  becomes the weight of all sampled plans to the

<sup>3</sup>These probabilities are well-defined:  $P(G) + P(\neg G) = 1$ , and  $P(O) + P(\neg O) = 1$ , where  $P(\neg O) := P(\hat{\mathcal{M}} \setminus C(O))$ ; calculating  $P(O, G) = P(O \mid G) \cdot P(G)$  and using  $P(\neg G)$  to obtain  $P(O, \neg G)$ , then  $P(O, G) + P(O, \neg G) = P(O)$ .

goal extending  $O$ , over the weight of all sampled maximal a-trajectories extending  $O$ :

$$P(G \mid O) = \frac{P(O \mid G) \cdot P(G)}{P(O)} = \frac{\sum_{\pi'' \in C_G(O)} w(\pi'')}{\sum_{\pi' \in C(O)} w(\pi')}. \quad (4)$$

Any set of a-trajectories for which probabilities  $P(O \mid G)$  and  $P(G \mid O)$  are defined for all elements in the set can then be ranked to favour trajectories that are more goal-intended according to either likelihoods or posteriors, respectively,

$$\arg \max_O P(O \mid G) = \arg \max_O \sum_{\pi' \in C_G(O)} w(\pi'), \quad (5)$$

$$\arg \max_O P(G \mid O) = \arg \max_O \frac{P(O \mid G)}{P(O \mid \neg G)}, \quad (6)$$

where equation 6 is obtained by simplifying  $\arg \max_O P(G \mid O) / P(\neg G \mid O)$  and noting that  $P(G)$  and  $P(\neg G)$  are constant when considering a single goal in planning problems. When extending the domain of conditional probabilities to the set of all possible observations in a planning problem, we adopt the convention of setting undefined probabilities to 0. This reflects an observer that assumes unknown trajectories are not goal directed.

### Framework Properties

We study the properties of a search guided by Equations 5 and 6. We first state the results, followed by analysis.

**Claim 1.** *Given non-empty  $\hat{\mathcal{M}}$  and  $\hat{\mathcal{M}}_G$ , any  $w$ , and a-trajectories  $O_e$  extending an a-trajectory  $O_p$  by one action,  $P(O_e \mid G) \leq P(O_p \mid G)$ , as  $C_G(O_e) \subseteq C_G(O_p)$ .*

**Claim 2.**  *$P(O \mid G) = 0$  and  $P(G \mid O) = 0$  for all a-trajectories  $O$  that are not extended by any plan  $\pi' \in \hat{\mathcal{M}}_G$ .*

**Lemma 1.** *Given non-empty  $\hat{\mathcal{M}}$  and  $\hat{\mathcal{M}}_G$ , and any  $w$ ,  $\max_{O_e} P(G \mid O_e) \geq P(G \mid O_p)$ .*

*Proof sketch.* Every maximal a-trajectory extending  $O_p$  extends exactly one child  $O_e$ ; hence  $C(O_p) = \bigcup_{O_e} C(O_e)$  and  $C_G(O_p) = \bigcup_{O_e} C_G(O_e)$ , and these unions are disjoint. Let  $X(O) = \sum_{\pi' \in C(O)} w(\pi')$  and define  $X_G(O)$  analogously. Then  $X(O_p) = \sum_{O_e} X(O_e)$  and  $X_G(O_p) = \sum_{O_e} X_G(O_e)$ , so

$$P(G \mid O_p) = \frac{X_G(O_p)}{X(O_p)} = \sum_{O_e} \frac{X(O_e)}{X(O_p)} \frac{X_G(O_e)}{X(O_e)}.$$

Thus  $P(G \mid O_p)$  is a weighted average of its children and therefore cannot exceed all of them.  $\square$

**Theorem 1.** *Given non-empty  $\hat{\mathcal{M}}$  and  $\hat{\mathcal{M}}_G$ , and any  $w$ , a planner that expands  $\max P(G \mid O)$  and breaks ties by greater trajectory length, will find a plan in number of expansions  $m \leq \max_{\pi' \in \hat{\mathcal{M}}_G} |\pi'|$ .*

*Proof.* The first expanded node has  $P(G \mid O)$  greater than or equal to all others. Since  $P(G \mid O) > 0$ , Claim 2 ensures that at least one plan to the goal exists, and by Lemma 1 its best child has probability  $\geq$  that of its parent. Since it also has greater trajectory length, it is expanded next; induction completes the argument.  $\square$

**Lemma 2.** Let  $|\hat{\mathcal{M}}_G| > 0$ . Suppose  $w$  is a weight function such that  $w(\pi) > w(\pi') \iff \text{cost}(\pi) < \text{cost}(\pi')$ . For a planner that expands a-trajectories in order of  $\max P(O | G)$ , the first expanded goal node is guaranteed to be a minimal-cost plan among all plans in the sample  $\hat{\mathcal{M}}$ . The result also holds under the weaker condition  $\text{cost}(\pi) \leq \text{cost}(\pi') \Rightarrow w(\pi) \geq w(\pi')$ , provided ties in  $P(O | G)$  are broken by preferring shorter trajectories.

*Proof sketch.* Assume the first goal-reaching a-trajectory  $\pi$  expanded is not minimal cost, and let  $\pi^*$  be a cheaper plan. If  $\pi^*$  is fully generated, then  $P(\pi^* | G) > P(\pi | G)$  since  $w(\pi^*) > w(\pi)$ , contradicting that  $\pi$  was expanded first. If instead only its prefix  $\rho$  is generated, then  $P(\rho | G) \geq P(\pi^* | G)$  because prefixes aggregate the weight of all their extensions and these values decrease monotonically with depth. Thus  $\rho$  (or  $\pi^*$  when complete) would have been expanded before  $\pi$ . Hence the first expanded goal-achieving trajectory must correspond to a minimal-cost plan. When different-cost plans have equal weight, tie-breaking by shorter length selects the minimal-cost one first.  $\square$

**Lemma 3.** A planner that expands a-trajectories in order of  $\max P(O | G)$  will expand at most  $\sum_{\pi \in \hat{\mathcal{M}}_G} (|\pi| - 1) + 1$  nodes before expanding a goal, and  $\sum_{\pi \in \hat{\mathcal{M}}_G} (|\pi| - 2) + 1$  nodes before generating a goal.

*Proof.* Follows from Claim 2 that at most all non-goal nodes in s-trajectories implied by plans in  $\hat{\mathcal{M}}_G$  will be expanded before expanding a goal node. In the worst case, plans in  $\hat{\mathcal{M}}_G$  do not overlap and only share the initial state, which is counted once.  $\square$

**Theorem 2.** Given non-empty  $\hat{\mathcal{M}}$  and  $\hat{\mathcal{M}}_G$ . For a planner that expands a-trajectories in order of  $\max P(O | G)$ ; as samples are added to  $\hat{\mathcal{M}}_G$ , the length of the first expanded plan is non-increasing, and the worst-case number of expansions is non-decreasing.

*Proof.* Follows from previous Lemmas 2 and 3, and considering that as samples are added, the minimum plan cost in the set can only decrease.  $\square$

**Remarks.** We briefly summarise general properties derived from the presented theorems. Claim 2 implies that following any a-trajectory with both  $P(O | G) > 0$  and  $P(G | O) > 0$  is a valid strategy for reaching a goal. Theorem 1 shows that expanding nodes according to Equation 6 follows a hill climbing strategy when ties are broken by larger  $g$  and is strongly goal directed. If ties are broken by smaller  $g$ , then it may perform local searches when ties are encountered, until it finds an exit to the plateau. In contrast, Equation 5 follows the maximum a posteriori path, inducing an exploratory strategy akin to an A\* search with a consistent heuristic, as shorter a-trajectories tend to have higher  $P(O | G)$ , noted in Claim 1. This approach expands sampled solution trajectories until it selects a sample optimal plan. Theorem 2 reflects the sampling exploration-exploitation trade-off for  $P(O | G)$ : increasing the number

of samples in  $\hat{\mathcal{M}}_G$  can improve solution quality, but also increases the worst case number of expansions, reflecting the larger exploratory effort required.

## Uniform Regimes

In what follows, we introduce and analyze the properties of two basic weight functions. We consider these as the two general uniform weighting processes, where we assign equal probability to, respectively (1) every sampled maximal a-trajectory, (2) every action choice in state transitions.

We define a **Uniform Maximal a-trajectory Probability (UMP)** weight function as a weight function  $w(\pi) = c$  where  $c$  is a positive constant, implying a uniform preference of the agent towards any sampled maximal a-trajectory in  $\hat{\mathcal{M}}$ . Let us define quantities  $N_T = |\hat{\mathcal{M}}|$ ,  $N_G = |\hat{\mathcal{M}}_G|$ ,  $N_C(O) = |C(O)|$ , and  $N_{CG}(O) = |C_G(O)|$ .

**Corollary 1.** Given non-empty  $\hat{\mathcal{M}}$  and  $\hat{\mathcal{M}}_G$ , and a UMP weight function  $w$ , the probabilities obtained become  $P(O) = \frac{N_C(O)}{N_T}$ ,  $P(G) = \frac{N_G}{N_T}$ ,  $P(O | G) = \frac{N_{CG}(O)}{N_G} \propto N_{CG}(O)$ ,  $P(G | O) = \frac{p(O|G)p(G)}{p(O)} = \frac{N_{CG}(O)}{N_C(O)}$ .

*Proof.* Follows from Equations 1,2,3, and 4; setting weight function  $w(\pi) = 1$ , then the value of each summation is equivalent to the number of elements in the relevant sets.  $\square$

Corollary 1 shows that under a UMP weight function,  $P(O | G)$  is proportional to the number of plans extending  $O$ , and ordering the open list by  $\arg \max P(O | G)$  favours such prefixes. Ordering by  $\arg \max P(G | O)$  favours prefixes with a higher ratio of plan completions to non-plan continuations. Both quantities can be seen as *measures of robustness* of a partial solution, biasing the search towards directions with more valid outcomes.

A **Uniform Transition Probability (UTP)** weight function assigns to a maximal I-reachable a-trajectory  $\pi = \langle a_0, \dots, a_{k-1} \rangle$  the product of uniform action probabilities at each step,  $w(\pi) = \prod_{i=0}^{k-1} [|\mathbf{A}(s_i)|]^{-1}$ , where  $s_0$  is the initial state,  $s_{i+1} = f(a_i, s_i)$ , and  $\mathbf{A}(s_i)$  is the set of applicable actions at state  $s_i$ . That is, at each step the agent selects an applicable action with uniform probability.

**Lemma 4.** Given non-empty  $\hat{\mathcal{M}}$  and  $\hat{\mathcal{M}}_G$ , a UTP weight function  $w$ , and an a-trajectory  $O$  that is extended by single solution plan  $\pi_s$ , the number of nodes generated to find  $\pi_s$  by a planner that expands according to  $\max P(O | G)$  is lower bounded by  $-\ln [P(O | G) \cdot P(G)] \cdot e \propto -\ln [P(O | G)]$ .

*Proof.* The lower bound on nodes generated is given by the minimum possible number of nodes generated while following  $\pi_s$  that achieves UTP weight  $w(\pi_s) = P(O | G) \cdot P(G)$ . For each expanded state  $s_i \in \mathcal{T}(\pi_s)$ , the number of generated nodes increases by  $|\mathbf{A}(s_i)|$ , and the weight of the a-trajectory to  $s_i$  is multiplied by  $\frac{1}{|\mathbf{A}(s_i)|}$ . For  $w(\pi_s) = \frac{1}{X}$  the minimum number of generated nodes is thus given by solving  $\min \sum_{s_i \in \mathcal{T}(\pi_s)} (|\mathbf{A}(s_i)|) \text{ s.t. } \prod_{s_i \in \mathcal{T}(\pi_s)} |\mathbf{A}(s_i)| = X$ . A lower bound to the integer solution is achieved by solving the real version of the problem, which can be solved analytically through the AM-GM Inequality to yield  $e \cdot \ln(X)$ .  $\square$

**Theorem 3.** *Given non-empty  $\hat{\mathcal{M}}$  and  $\hat{\mathcal{M}}_G$ , a UTP weight function  $w$ , and a planner that expands according to  $\max P(O | G)$ , a lower-bound number of node generations required to achieve any plan that extends  $O$  is  $-\ln[P(O | G) \cdot P(G)] \cdot e$ .*

*Proof.*  $P(O | G)$  is equivalent to the sum of the weight of all plans extending  $O$ , over a common denominator. From Lemma 4, it follows that the number of node generations required to solve any plan increases inversely to the plan's weight. Thus, the minimum number of node generations occurs when a single solution plan extends  $O$ .  $\square$

Theorem 3 shows that expanding nodes according to Equation 5 with a UTP weight function follows the a-trajectory that minimises the best-case number of node generations to find a plan in  $\hat{\mathcal{M}}$ . This strategy can be seen as optimistic in the face of uncertainty, where uncertainty refers to unexplored regions of the state space. It assumes the sub-graph extending the selected trajectory has an ideal shape; as new information is revealed, this estimate may worsen, leading the search to prefer other sub-graphs. Such a bound cannot be obtained with a UMP weight function, as uniform weights are not tied to the number of node generations.

### Estimating Measures of Goal Intention

The IRPL model helps us characterize intention-based heuristics and their search behaviour. The prior knowledge we have assumed so far is, however, unrealistic, as we cannot expect to have access to a maximal trajectory library  $\hat{\mathcal{M}}$  and plan library  $\hat{\mathcal{M}}_G$ . To obtain a practically relevant framework, we need to account for an observer that estimates goal intentionality using approximate measures.

Existing work in GR-as-planning defines the likelihood  $P(O | G)$  through the use of cost estimates (Ramirez and Geffner 2010; Masters and Sardiña 2021). It assumes that rational agents are more likely to prefer lower-cost plans to their selected goal, and consequently relies on the sub-optimality of plans to each goal that incorporate observed events to estimate goal intentionality. We avoid following this direction, because adopting such formulations would lead us back to using (heuristic) cost estimates to direct the search. Instead, we propose a novel formulation for approximating and interpreting goal intentionality which provides: 1) a theoretical background for alternative directions for estimating heuristics in classical planning, and 2) a generalisation of the IRPL model in terms of a larger class of approximate divergence-based goal-intended models.

### General Model of Divergence-based Likelihood

We define a divergence-based generalisation of goal-intended likelihood through Equation 7,

$$\tilde{P}(O | G) := e^{-D_{\text{KL}}(d_{O,G} \| d_G)}, \quad (7)$$

where  $d_G$  is a distribution that estimates goal-intended outcome probabilities (intuitively capturing likelihoods of outcomes in  $\mathcal{M}_G$ ,  $P(\pi | G)$ ), and  $d_{O,G}$  denotes a specific posterior distribution over the same space obtained by incorporating observed evidence (intuitively capturing updated likelihoods of outcomes given observation  $O$ ,  $P(\pi | O, G)$ ).

It is important to keep in mind that observations may only reveal outcomes for a subset of the domain variables. Thus, to illustrate the probabilistic intuition underlying the adopted distributions, we provide the following simplified example of rolling a dice 10 times, where the goal  $G$  is a total sum of values  $> 30$ . A complete outcome is the sequence  $r = (r_1, \dots, r_{10})$  of the random variable  $R = (R_1, \dots, R_{10})$ , where each  $r_i$  denotes the value of one roll, and the distribution  $d_G := \rho(r | G)$  assigns probabilities to all such sequences. An observation reveals the values of a subset of these variables, e.g.,  $(R_1 = 3, R_2 = 5)$ , and observations are non-exclusive: observing  $(3, 5)$  does not exclude observing  $(3, 5, 4)$  as a longer trace. The observation instead belongs to the marginal domain induced by the variables it reveals, with the marginal distribution  $\rho(r_1, r_2 | G)$  assigning probabilities to the exclusive outcomes of that sub-domain. The posterior distribution then incorporates observations:  $d_{O,G} := \rho(r | R_1 = 3, R_2 = 5, G)$ .

Furthermore, in the IRPL model observations are defined as i-reachable a-trajectories, whereas now the domain of  $d_G$  defines the outcomes that determine the events in observations. To illustrate this, following the previous example, assume each roll outcome  $r_i$  no longer represents the value of the dice roll, but rather whether the roll gives a number  $> 3$ . Observations would then take the form  $(R_1 = \text{true}, R_2 = \text{false}, \dots)$ . While approximate, such events still contain useful information for estimating goal achievement.

A well-defined KL-divergence requires matching the domains of  $d_{O,G}$  and  $d_G$ . Since the notation  $O$  does not specify the domain of the observation, we instead adopt  $x \in \mathcal{X}$ ,  $y \in \mathcal{Y}$ , and  $z \in \mathcal{Z}$  to denote outcomes explicitly associated with their domains. Let  $\mathcal{X} = \mathcal{X}_1 \times \dots \times \mathcal{X}_v$  be the domain of complete outcomes, with  $x = (x_1, \dots, x_v)$  denoting a full outcome. For any index sets  $\mathcal{J} \subseteq K \subseteq \{1, \dots, v\}$ , define the marginal domains  $\mathcal{Y} = \times_{i \in \mathcal{J}} \mathcal{X}_i$  and  $\mathcal{Z} = \times_{i \in K} \mathcal{X}_i$ , with tuples  $y = (x_i)_{i \in \mathcal{J}}$  and  $z = (x_i)_{i \in K}$ ,  $x_i \in \mathcal{X}_i$ , denoting outcomes in  $\mathcal{Y}$  and  $\mathcal{Z}$ . We write  $\rho(x)$  for the distribution over complete outcomes, and  $\rho(y)$  and  $\rho(z)$  for the corresponding marginals. Let  $y^* \in \mathcal{Y}$  be the discrete deterministic observation  $O$ . We add superscripts to specify the domain of  $d_{O,G}$  and  $d_G$ . Let  $d_G^{\mathcal{X}} := \rho(x | G)$  and  $d_{O,G}^{\mathcal{Y}} := \rho(y | y^*, G)$ . Let  $d_G^{\mathcal{Z}}$  and  $d_{O,G}^{\mathcal{Z}}$  denote any domain-matched representation of  $d_G^{\mathcal{X}}$  and  $d_{O,G}^{\mathcal{Y}}$  on an intermediate domain  $\mathcal{Z}$ .  $d_G^{\mathcal{Z}}$  is marginalisation  $\rho(z | G) = \frac{\rho(z|x,G) \cdot \rho(x|G)}{\rho(x|z,G)}$ ;  $d_{O,G}^{\mathcal{Z}}$  is Bayesian update  $\rho(z | y^*, G) = \frac{\rho(y^*|z,G) \cdot \rho(z|G)}{\rho(y^*|G)}$ .

**Theorem 4.** *The KL divergence is invariant to the choice of  $\mathcal{Z}$ , and coincides with the negative log-likelihood of the observation conditional on the goal:*

$$D_{\text{KL}}(d_{O,G}^{\mathcal{Z}} \| d_G^{\mathcal{Z}}) = -\log \rho(y^* | G).$$

*Proof.* Expanding  $\rho(z | y^*, G)$ , simplifying, and marginalising, we have

$$\begin{aligned} D_{\text{KL}}(d_{O,G}^{\mathcal{Z}} \| d_G^{\mathcal{Z}}) &= \sum_{z \in \mathcal{Z}} \rho(z | y^*, G) \log \frac{\rho(z | y^*, G)}{\rho(z | G)} \\ &= \sum_{z \in \mathcal{Z}} \left[ \rho(z | y^*, G) \log \rho(y^* | z, G) \right] - \log \rho(y^* | G). \end{aligned}$$

Since  $y^*$  is a projection of some  $z$  onto observed coordinates, for all  $z \in \mathcal{Z}$  if any variable value in  $y^*$  is different from the corresponding variable in  $z$ ,  $\rho(z | y^*, G) = 0$ , and if all variables in  $y^*$  have same value with corresponding variables in  $z$ ,  $\rho(y^* | z, G) = 1$ , so  $\log \rho(y^* | z, G) = 0$ . Thus,

$$D_{\text{KL}}(d_{O,G}^{\mathcal{Z}} \| d_{O,G}^{\mathcal{Z}}) = -\log \rho(y^* | G). \quad \square$$

Through Theorem 4, we thus show that the specific choice of intermediate domain used to align distributions, required for computing the KL-divergence, does not affect the derivation of  $-\log \rho(y^* | G)$ . This justifies Equation 7, and points at multiple possible strategies to estimate this quantity.

**Special cases.** We briefly present cases of interest:

1) *Marginalising  $d_G^{\mathcal{X}}$  to  $\mathcal{Y}$ .* We can make  $d_G^{\mathcal{X}}$  match the marginal distribution of  $d_{O,G}^{\mathcal{Y}}$  by marginalising all domain factors not in  $\mathcal{Y}$ , directly obtaining likelihood  $d_G^{\mathcal{Y}} := \rho(y | G) = \frac{\rho(y|x,G) \cdot d_G^{\mathcal{X}}}{\rho(x|y,G)} = \sum_x \rho(y, x | G)$ ;  $d_{O,G}^{\mathcal{Y}}$  is then a degenerate PMF  $\rho(y | y^*, G)$ , i.e., a one-hot distribution over its domain, assigning 1 to observation  $y^*$ , and 0 to all others.

2) *Mapping  $d_{O,G}$  to  $\mathcal{X}$ .* We can map the observation distribution to the full domain  $\mathcal{X}$  through Bayesian update  $d_{O,G}^{\mathcal{X}} := \rho(x | y^*, G) = \frac{\rho(y^*|x,G) \cdot \rho(x|G)}{\rho(y^*|G)}$ .

3) *Consistency with IRPL Model.* When  $d_G^{\mathcal{X}}$  perfectly describes the plan library, through a bijective mapping of every sampled plan to a possible outcome with its relative weight as probability, and  $d_{O,G}^{\mathcal{Y}}$  describes an observed I-reachable a-trajectory, Equation 7 then derives the IRPL likelihood. It thus generalises Equation 3. Details are included in Appendix A of the extended version (Rosa et al. 2026).

**Assumptions.** The result holds whenever (i) the observation is discrete and deterministic over  $\mathcal{Y}$ , i.e.,  $d_{O,G}^{\mathcal{Y}}(y) = \mathbf{1}[y = y^*]$ , and (ii) the observation map  $m : \mathcal{X} \rightarrow \mathcal{Y}$  is surjective onto its image, ensuring that at least one complete trace is consistent with  $y^*$ . These are consistent with the GR model.

**Interpretation.** The cost based formulation of goal recognition (Ramirez and Geffner 2010; Masters and Sardiña 2021) is grounded in the assumption of agent rationality: observations that align with near optimal plans are considered evidence of goal-intended behaviour. Similarly, Equation 7 interprets rationality in terms of the divergence of observed information  $d_{O,G}$  from the distribution  $d_G$ , which captures statistical evidence of goal-intendedness from estimated solution paths. It is intuitive that observations consistent with goal-intended behaviour, when normalised by the likelihood of the observation, reflect intent towards a goal. In our experiments, we show that estimating such information can provide a signal for informing an efficient state-space exploration and traversal, whose behaviour still correlates with our results for the ideal but impractical IRPL model.

## Planning Heuristics

We demonstrate the practical relevance of our conceptual framework by proposing two variants of a novel intention-based heuristic which align with our theory, allowing us to

correlate results with our predictions of search behaviour obtained through analysis of the IRPL model. We follow recent work in GR (Pereira, Oren, and Meneguzzi 2017; Wilken et al. 2024), which represents observations through the facts implied by underlying action sequences or trajectories. In a GR setting, this would represent facts added by actions in action sequence  $O$ . In a planning context, it may also include facts present in the initial state.

Our heuristic uses *fact observation probability estimation* as described in (Wilken et al. 2024) to estimate, for each fact  $q \in F$ , a Bernoulli distribution  $q \sim \text{Bernoulli}(P_{\text{rel}}(q | G))$ , representing the probability that  $q$  is achieved at any point in a *delete-relaxed* plan.  $P_{\text{rel}}(q | G)$  then refers to the conditional probability in the relaxed problem. This is done by first sampling a set of delete-relaxed plans to the goal to obtain the probability of observing each action in a sampled relaxed plan. The probability of observing facts is then indirectly derived as the probability of not observing any supporter action for the fact in a relaxed plan; thus, even if a fact is supported in all sampled relaxed plans, it may still get an observation probability less than 1. Pseudocode for our implementation is included in the extended version (Rosa et al. 2026). Wilken et al. (2024) adopt the technique in GR problems, using fact observation probabilities to weight vectors in a feature space, and then using the Euclidean norm of the vectors to estimate the distance covered by an agent towards a goal. Our divergence-based interpretation of goal-intendedness instead informs us to use fact observation probabilities to directly estimate the likelihood.

**Likelihood Estimation.** Let  $O^F$  represent a set of observed facts. If we assume conditional independence between fact occurrences, the likelihood is simply

$$\tilde{P}(O^F | G) = \prod_{q \in O^F} P_{\text{rel}}(q | G).$$

According to our model, this likelihood can be interpreted in terms of a Boltzmann distribution of the marginalisation of  $d_G := \tilde{\rho}(q_1, \dots, q_{|F|}) = \prod_{q \in F} \text{Bernoulli}(P_{\text{rel}}(q | G))$  with respect to facts in  $O^F$ . It represents the divergence between distribution  $\tilde{\rho}(q_1, \dots, q_{|F|})$  and the observation  $O^F$ , and is thus in the same class of divergence-based measures as the IRPL likelihood. This aspect justifies our comparison of solver behaviour with theory from our IRPL analysis.

In practice, we calculate the relative ranking of observations through the sum of log-probabilities. As the fact occurrence probability computation is only performed once at the beginning of the search (at the initial state),  $\tilde{P}(O^F | G)$  is then computable in time linear in  $|O^F|$  during the search, providing quick evaluation. In our experiments, we set the number of sampled relaxed plans to 100. This value was selected through analysis of results in the context of GR in (Wilken et al. 2024), and empirical testing. We refer to this heuristic as *relaxed plan observation likelihood* ( $ol^{\text{rp}}$ ).

**Posterior Estimation.** A Bayesian posterior is also derived from  $P(G | O) \propto_{\text{rank}} \frac{P(O|G)}{P(O|\neg G)}$  (from Equation 6):

$$\tilde{P}(G | O^F) \propto_{\text{rank}} \sum_{q \in O^F} \log P_{\text{rel}}(q | G) - \log P_{\text{rel}}(q | \neg G).$$

This requires an additional estimation of  $P_{\text{rel}}(q \mid \neg G)$ , which uses a modification of the procedure used to obtain  $P_{\text{rel}}(q \mid G)$ : rather than sampling delete-relaxed plans, the heuristic samples delete-relaxed action sequences that achieve all the achievable non-goal facts in the instance. Fact probabilities  $P_{\text{rel}}(q \mid \neg G)$  are then extracted from these sets of supporter actions. We still sample 100 fact sets each for both  $P_{\text{rel}}(q \mid G)$  and  $P_{\text{rel}}(q \mid \neg G)$ , and call this variant *relaxed plan observation posterior* ( $op^{rp}$ ).

### Adding Goal-intentionality to BFWS

We integrate our proposed heuristics with a BFWS solver (Lipovetzky and Geffner 2017), which balances exploration of the search space and exploitation of heuristics, to evaluate improvements in Table 1. BFWS( $f5$ ) uses *Partitioned Novelty* (Lipovetzky and Geffner 2017) as primary heuristic to partition each planning problem into multiple sub-problems and prioritize exploration, and the *goal-count heuristic*  $h^{GC}$ , that counts the number of unachieved goal facts, is used both to inform such partitioning, and as tie-breaking heuristic. These heuristics are quickly computable, but often not very granular, creating a large number of ties. Minimum distance  $g$  is used to break final ties, and has a big impact on the overall performance of the planner. We implement our proposed variants as third tie-breakers after  $h^{GC}$  to provide a fast but more informed tie-breaking mechanism that seeks progress by directing the search towards the goal. Our aim is for this to lead to states that improve the other heuristics more effectively than  $g$ . According to the IRPL model analysis,  $ol^{rp}$  is supposed to exhibit an A\*-like exploratory behaviour, augmenting the original tie-breaking mechanism, whereas  $op^{rp}$  should induce a more greedy and goal-directed search.

**Partial Observation Trajectories.** We adapt  $ol^{rp}$  and  $op^{rp}$  to BFWS through  *$ol^{rp}$ -restart* ( $ol_r^{rp}$ ) and  *$op^{rp}$ -restart* ( $op_r^{rp}$ ). These adaptations add log probabilities from facts that have been observed only in the trajectory from the last state that improved  $h^{GC}$ , as opposed to all facts achieved from the start. By restarting at the most recent  $h^{GC}$  improvement, it regains informedness in the cases where it was lost. Otherwise, if a fact that did not appear in any sampled relaxed plan to the goal is observed, it would strongly penalise the probability of all descendant nodes. This helps account for inaccuracies in fact occurrence estimates introduced by adopting distributions derived from relaxed plans.

## Experimental Results

We run our experiments on an AMD EPYC 7763 processor, with each test running on a single core. We adopt Downward Lab’s experiment module (Seipp et al. 2017), whereas our proposed solvers and heuristics are implemented in C++ using the LAPKT planning module (Ramirez et al. 2015). Our adopted branch of LAPKT uses the FD grounder (Helmert 2009), with the exception of problems that produce axioms, which are not currently supported in LAPKT. In such problems, our planners automatically fallback to the Tarski grounder (Francés, Ramirez, and Collaborators 2018; Singh et al. 2023). All experiments are limited to 1800 seconds

time and 8 GB memory constraints, following the satisficing track of the *International Planning Competition* (IPC) (Taitler et al. 2024). The problem set is composed of all IPC satisficing track benchmarks, selecting the latest problem sets for recurrent domains.

**Experimental Analysis.** Table 1 highlights the effect of incorporating proposed heuristics into BFWS( $f5$ ). Both  $f5\text{-}ol_r^{rp}$  and  $f5\text{-}op_r^{rp}$  improve coverage and reduce the number of expansions on a significant portion of problems compared to the BFWS( $f5$ ) baseline. This confirms that the goal-intendedness encoded by our heuristics induces a more efficient exploration of the state space.  $f5\text{-}op_r^{rp}$  shows a small advantage over  $f5\text{-}ol_r^{rp}$  in node expansions, aligning with our IRPL analysis that  $P(G \mid O)$  drives a more goal directed greedy search. However, the more exploratory nature of  $P(O \mid G)$  leads  $f5\text{-}ol_r^{rp}$  to solve more instances overall. In regards to the exploration-exploitation dilemma, it is reasonable to expect that the effectiveness of exploitation declines as the quality of the underlying estimates worsens. Thus, the more exploratory heuristic may be better suited to handle the approximation noise. The additional overhead from computing fact observation probabilities is offset by improved search efficiency, resulting in lower average solution times than BFWS( $f5$ ). By contrast, the extra computation of  $P(O \mid \neg G)$  required by  $op_r^{rp}$  introduces a small penalty relative to  $ol_r^{rp}$ , but differences are within the measurements’ standard deviation. The main drawback of both techniques is their effect on plan cost, which is expected due to the bias toward shorter plans induced by the original  $g$  tie-breaker. Suboptimal plan costs may also be influenced by the trajectory-restart policy of  $ol_r^{rp}$ , which loses the global negative bias towards shorter trajectories from Claim 1.

**Enhanced Variants.** We also evaluate enhanced versions of the proposed solvers from Table 1 in Table 2. *BFWS( $f5$ )-Landmarks- $ol_r^{rp}$*  ( $OL_r^{rp}$ ) substitutes  $h^{GC}$  with the *Landmarks heuristic* (Porteous, Sebastia, and Hoffmann 2001), and uses a *single trimmed open list* (Rosa and Lipovetzky 2024) for greater memory efficiency. Both the coverage and average solution time gap between  $ol_r^{rp}$  and  $op_r^{rp}$  widen in this configuration (see Rosa et al. 2026), hence we do not include the latter. We also test variant  *$OL_r^{rp}$ -UTP* with modified fact observation probability sampling, to highlight the practical impact of the weighting scheme in the IRPL model. When calculating supporter action probabilities for  $ol_r^{rp}$ , each sampled relaxed plan is given an equal weight.  $OL_r^{rp}$ -UTP re-weights sampled relaxed plans according to a UTP weight function, giving more importance to relaxed plans with greater UTP weight. Lastly, we provide a comparison with recent high-coverage dual strategy solvers – that run an initial search and, if this fails, fall back to a backend solver – by replacing the frontend solver in BFNoS-Dual (Rosa and Lipovetzky 2024) with  $OL_r^{rp}$  ( *$OL_r^{rp}$ -Dual*).

**Results.** Table 2 compares enhanced variants with multiple SoTA Novelty planners *BFNoS-Dual* (Rosa and Lipovetzky 2024), *Dual-BFWS* (Lipovetzky and Geffner 2017), and *Approximate Novelty Tarski* (Singh et al. 2021), as well as past IPC satisficing track winners *LAMA* (Richter and Westphal

	$f5$	$f5\text{-ol}_r^{TP}$	$f5$	$f5\text{-op}_r^{TP}$	$f5\text{-ol}_r^{TP}$	$f5\text{-op}_r^{TP}$
Coverage (1831)	1,510	1,560 (5.03)	1,510	1,556 (1.5)	1,560 (5.03)	1,556 (1.5)
% Score	76.77%	80.20% (0.35)	76.77%	79.90% (0.15)	80.20% (0.35)	79.90% (0.15)
N Fewer Expansions	368.2 (6.7)	1,134.4 (8.7)	341.4 (9.1)	1,159.4 (8.0)	412.0 (7.3)	716.6 (6.5)
N Lower Time	582.4 (99.5)	897.0 (92.2)	658.8 (117.3)	814.6 (102.4)	811.8 (130.1)	658.0 (122.6)
N Lower Plan Cost	760.6 (8.9)	369.6 (7.6)	763.6 (2.9)	375.8 (1.7)	375.2 (4.5)	349.2 (13.4)
Average EpS	40,111	35,680	40,111	33,626	35,680	33,626

Table 1: Enhancing a basic BFWS( $f5$ ) solver with proposed  $ol_r^{TP}$  and  $op_r^{TP}$  heuristics. **% score** is the average of the % of instances solved in each problem domain. Results for comparisons and solvers with a randomised component represent the mean, and include the standard deviation across 5 measurements. *N Fewer Expansions*, *N Lower Time* and *N Lower Plan Cost* represent the number of problems where a variant scores better in the respective metrics (ignoring ties). Avg. EpS represents the average number of expansions per second across problems solved by all planners, and proxies computational overhead. Results indicate that  $ol_r^{TP}$  and  $op_r^{TP}$  reduce the number of expansions across a significant portion of problems, on average improving also coverage and solution time. Such improvements come at the cost of on average worse plan quality compared to the baseline.

Planner	Coverage (1831)	% score	Agile score
Dual-BFWS	1,607	83.6%	1,200.8
ApxNoveltyT	1,611 (3.5)	83.8% (0.2)	1,233.7 (0.2)
LAMA	1,535	79.1%	1,192.3
Scorpion-M	1,591	82.9%	1,206.4
$OL_r^{TP}$	1,621 (3.2)	84.6% (0.3)	1,229.4 (3.4)
$OL_r^{TP}$ -UTP	1,616 (2.1)	84.2% (0.1)	<b>1,236.0 (1.9)</b>
BFNoS-Dual	1,641 (0.6)	86.2% (0.1)	1,173.3 (3.5)
$OL_r^{TP}$ -Dual	<b>1,655 (1.5)</b>	<b>87.0% (0.1)</b>	1,232.6 (2.9)

Table 2: Mean coverage and Agile score of enhanced variants. Our enhancements of BFWS( $f5$ ) outperform SoTA planners, on average solving more problems, more quickly, without using multiple open lists or runs.  $OL_r^{TP}$ -Dual provides an improved dual-strategy solver, solving more problems than all other tested planners and attaining a meaningfully higher Agile score than BFNoS-Dual, the second-best in coverage.

2010) and *Scorpion-Maidu* (Corrêa et al. 2023). Results indicate improved coverage of our proposed modifications of BFWS( $f5$ ).  $OL_r^{TP}$ -UTP also improves **Agile score**<sup>4</sup> over the base  $OL_r^{TP}$ , at the cost of some problem coverage.

We further note a correlation between our experimental results for  $OL_r^{TP}$ -UTP and Theorem 3, suggesting that prioritising sampled relaxed plans with tighter estimated bounds on the number of node generations to reach the goal can accelerate search, albeit at the expense of coverage, as it may reduce the robustness that comes with using uniform weights discussed in Corollary 1. UTP weights promote a more committed search, whereby a focus on optimistic relaxed plan samples can lead to earlier solutions when these estimates are accurate, but may also mislead the search when they are not. The latter case arises when sampled relaxed plans diverge from valid plans, potentially omitting important actions or facts, or including unhelpful ones.

<sup>4</sup>**Agile score** is a performance metric that jointly evaluates coverage and runtime. It assigns a score of 1 for problems solved in time  $T \leq 1s$ , and  $1 - \frac{\log(T)}{\log(300)}$  for  $1s < T \leq 300s$ . The total score is summed across all problems.

As fact probability extraction relies on the sampling of relaxed plans, we expect the quality of approximations to align with that of well-studied relaxed heuristics. For instance, our planners perform strongly on Settlers (Long and Fox 2003), a challenging domain where relaxed-plan-heuristic planners LAMA and Maidu outperform BFWS baselines (see Rosa et al. 2026). Moreover, sets of relatively poor plan-cost cases for variants in Table 1 appear to be domain-specific. We do not exclude that additional mechanisms related to the probability estimation process, such as observing sets of facts rather than actions, may impact per-domain amenability, informing future directions to improve proposed heuristics.

## Concluding Remarks

We introduce a model that provides an intuitive description of goal-intendedness given a set of underlying plans in an instance, and use it to study the behaviour of planners that adopt goal intentionality as heuristic signal. Our model and proposed heuristics are shown to belong to a new class of divergence-based goal recognition estimates. As a proof of concept, we propose two heuristics which improve the base performance of BFWS, matching complex IPC planners in Agile scores, while exceeding their coverage. We provide empirical evidence of correlation between our experimental results and the properties of our simplified theoretical model.

Our proposed planning-as-goal-recognition theoretical framework offers a new perspective on heuristic search, interpreting evaluation functions as processes that infer the intention of trajectories to the current state with respect to the goal. Trajectories, therefore, do not only reveal the cost so far, but also their goal-intendedness. This aspect can inform the design of new heuristics, and may provide a novel lens for analysing the properties of well-established classical planning heuristics. The one-off time cost of the information extraction phase in the proposed heuristics opens the door to more informed and expensive estimation methods to further improve problem solving capability. The probabilistic nature of the intentionality framework can also lead to new solutions in related problems, such as learned heuristics and hybrid planning, and facilitate connections between planning, goal recognition, and non-symbolic fields of AI.

## Acknowledgments

We thank the reviewers for their valuable feedback. Giacomo Rosa is supported by the Melbourne Research Scholarship established by the University of Melbourne. This research was supported by use of the Nectar Research Cloud and by the Melbourne Research Cloud. The Nectar Research Cloud is a collaborative Australian research platform supported by the NCRIS-funded Australian Research Data Commons (ARDC).

## References

- Bonet, B.; and Geffner, H. 2001. Planning as heuristic search. *Artificial Intelligence (AIJ)*, 129(1-2): 5–33.
- Bratman, M. E.; Israel, D. J.; and Pollack, M. E. 1988. Plans and Resource-Bounded Practical Reasoning. *Computational Intelligence*, 4(3): 349–355.
- Corrêa, A. B.; Francès, G.; Hecher, M.; Longo, D. M.; and Seipp, J. 2023. Scorpion Maidu: Width Search in the Scorpion Planning System. In *Proceedings of the 10th International Planning Competition (IPC-2023)*.
- Fikes, R. E.; and Nilsson, N. J. 1971. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence (AIJ)*, 2(3-4): 189–208.
- Francès, G.; Ramirez, M.; and Collaborators. 2018. Tarski: An AI Planning Modeling Framework. <https://github.com/aig-upf/tarski>.
- Helmert, M. 2009. Concise finite-domain representations for PDDL planning tasks. *Artificial Intelligence (AIJ)*, 173(5-6): 503–535.
- Hoffmann, J.; and Nebel, B. 2001. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research (JAIR)*, 14: 253–302.
- Kautz, H. A.; and Allen, J. F. 1986. Generalized plan recognition. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 32–37.
- Lipovetzky, N.; and Geffner, H. 2012. Width and serialization of classical planning problems. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, 540–545.
- Lipovetzky, N.; and Geffner, H. 2017. Best-first width search: Exploration and exploitation in classical planning. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, volume 31.
- Long, D.; and Fox, M. 2003. The 3rd international planning competition: Results and analysis. *Journal of Artificial Intelligence Research (JAIR)*, 20: 1–59.
- Masters, P.; and Sardiña, S. 2021. Expecting the unexpected: Goal recognition for rational and irrational agents. *Artificial Intelligence (AIJ)*, 297: 103490.
- Pereira, R.; Oren, N.; and Meneguzzi, F. 2017. Landmark-based heuristics for goal recognition. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, volume 31.
- Pollack, M. E. 1992. The Uses of Plans. *Artificial Intelligence (AIJ)*, 57(1): 43–68.
- Porteous, J.; Sebastia, L.; and Hoffmann, J. 2001. On the extraction, ordering, and usage of landmarks in planning. In *Proceedings of the European Conference on Planning (ECP)*.
- Ramirez, M.; and Geffner, H. 2009. Plan recognition as planning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1778–1783.
- Ramirez, M.; and Geffner, H. 2010. Probabilistic plan recognition using off-the-shelf classical planners. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 1121–1126.
- Ramirez, M.; Lipovetzky, N.; Singh, A.; and Muise, C. 2015. Lightweight Automated Planning ToolKiT. <http://lapkt.org/>.
- Richter, S.; and Westphal, M. 2010. The LAMA planner: Guiding cost-based anytime planning with landmarks. *Journal of Artificial Intelligence Research (JAIR)*, 39: 127–177.
- Rosa, G.; Honorio, J.; Lipovetzky, N.; and Sardiña, S. 2026. Goal Recognition as Planning: Deriving Heuristics from Intention Models - Extended Version. In *arXiv preprint arXiv:2603.14824*. <https://arxiv.org/abs/2603.14824>.
- Rosa, G.; and Lipovetzky, N. 2024. Count-based novelty exploration in classical planning. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, 4181–4189.
- Seipp, J.; Pommerening, F.; Sievers, S.; and Helmert, M. 2017. Downward Lab. <https://github.com/aibasell/lab>.
- Singh, A.; Lipovetzky, N.; Ramirez, M.; and Segovia-Aguas, J. 2021. Approximate novelty search. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, volume 31, 349–357.
- Singh, A.; Lipovetzky, N.; Ramirez, M.; Segovia-Aguas, J.; and Frances, G. 2023. Grounding Schematic Representation with GRINGO for Width-based Search. In *Proceedings of the 10th International Planning Competition (IPC-2023)*.
- Sukthankar, G.; Geib, C.; Bui, H. H.; Pynadath, D.; and Goldman, R. P. 2014. *Plan, activity, and intent recognition: Theory and practice*. Newnes.
- Taitler, A.; Alford, R.; Espasa, J.; Behnke, G.; Fišer, D.; Gimelfarb, M.; Pommerening, F.; Sanner, S.; Scala, E.; Schreiber, D.; et al. 2024. The 2023 International Planning Competition. <https://ipc2023.github.io/>.
- Torralba, A.; and Hoffmann, J. 2015. Simulation-Based Admissible Dominance Pruning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1689–1695.
- Wilken, N.; Cohausz, L.; Bartelt, C.; and Stuckenschmidt, H. 2024. Fact Probability Vector Based Goal Recognition. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, 4254–4261.