IMPLICIT BAYESIAN MARKOV DECISION PROCESS FOR RESOURCE-EFFICIENT EXPERIMENTAL DESIGN IN DRUG DISCOVERY

Tianchi Chen
Decision Science
Merck & Co., Inc.
Cambridge, MA, USA
tianchi.chen@merck.com

Jan Bima
Decision Science
MSD Czech Republic
jan.bima@msd.com

Otto Ritter
Decision Science
MSD Czech Republic
otto.ritter@msd.com

Sean L. Wu
Decision Science
Merck & Co., Inc.
San Francisco, CA, USA
sean.wu@merck.com

Bo Yuan
Pharmacokinetics, Dynamics, Metabolism, and Bioanalytical
Merck & Co., Inc.
San Francisco, CA, USA
bo.yuan@merck.com

Bingjia Yang
Pharmacokinetics, Dynamics, Metabolism, and Bioanalytical
Merck & Co., Inc.
San Francisco, CA, USA
bingjia.yang@merck.com

Xiang Yu
Pharmacokinetics, Dynamics, Metabolism, and Bioanalytical
Merck & Co., Inc.
West Point, PA, USA
xiang.yu@merck.com

ABSTRACT

In drug discovery, researchers make sequential decisions to schedule experiments, aiming to maximize probability of success towards drug candidates while simultaneously minimizing expected costs. However, such tasks pose significant challenges due to complex trade-offs between uncertainty reduction and allocation of constrained resources in a high-dimensional state-action space. Traditional methods based on simple rule-based heuristics or domain expertise often result in either inefficient resource utilization due to risk aversion or missed opportunities arising from reckless decisions. To address these challenges, we developed a Implicit Bayesian Markov Decision Process (IB-MDP) algorithm that constructs an implicit MDP model of the environment's dynamics by integrating historical data through a similarity-based metric, and enables effective planning by simulating future states and actions. To enhance the robustness of the decision-making process, the IB-MDP also incorporates an ensemble approach that recommends maximum likelihood actions to effectively balance the dual objectives of reducing state uncertainty and optimizing expected costs. Our experimental results demonstrate that the IB-MDP algorithm offers significant improvements over traditional rule-based methods by identifying optimal decisions that ensure more efficient use of resources in drug discovery.

1 INTRODUCTION

In drug discovery, strategic planning and selection of experiments play a pivotal role in impacting the pace and expenses of R&D activities. The identification of potential drug candidates requires conducting numerous assays at various stages of preclinical studies. The process often begins with limited information, creating significant challenges for achieving optimized outcomes due to time and budget constraints. Optimizing the use of resources to achieve targeted goals within these

limitations is among the most demanding tasks in creating effective Research Operation Plans (ROP). Conventional approaches, often relying on simple rule-based heuristics or domain expertise, struggle to adapt as new data emerges and typically fail to address state, model, and parameter uncertainties effectively, a challenge central to fields like Reinforcement Learning (RL) (Sutton & Barto, 2018) and Bayesian Experimental Design (BED) (Rainforth et al., 2024). Consequently, these methods often result in suboptimal decision-making and inefficient allocation of resources (Puterman, 2014).

To address these challenges, we propose the **Implicit Bayesian Markov Decision Process (IB-MDP)** algorithm, a *model-based* RL approach that constructs an implicit model of the dynamics of the environment by integrating historical data through a distance-based similarity metric (Bellet et al., 2013). Unlike traditional MDPs requiring explicit transition models (Puterman, 2014), or kernel RL methods often focused on value approximation (Ormoneit & Sen, 2002), IB-MDP uses similarity to drive a *generative* sampling process based on historical data (Alagoz et al.). This avoids precise parametrization and implicitly handles uncertainty about the underlying system state by dynamically adjusting data relevance, akin to belief updating in partially observable settings (see Appendix H for a conceptual POMDP framing), enabling efficient multi-step planning.

Moreover, to improve the robustness and reliability of decision-making, we incorporate an ensemble approach into the IB-MDP. While established for enhancing robustness in ML/RL (Dietterich, 2000; Zhou, 2012; Lakshminarayanan et al., 2017), we apply ensembles specifically to mitigate variance arising from IB-MDPś sampling-based model and MCTS planning (Osband et al., 2016). By aggregating policies from independent runs, we achieve more stable and reliable decision recommendations.

Our algorithm is demonstrated in the context of assay scheduling and ROP optimization, where it significantly improves resource utilization and decision quality compared to traditional heuristic-based approaches. The IB-MDP framework is broadly applicable to various resource-constrained decision-making tasks in drug discovery, making it a valuable tool for optimizing sequential decisions in preclinical studies.

Summary of Contributions: We introduce the IB-MDP, a model-based algorithm that integrates historical data using a distance-based similarity metric within the MDP framework, enabling efficient planning in sequential decision-making tasks. We incorporate an ensemble approach to enhance policy estimation, providing conceptual justification (Appendix H) and empirical evidence for its effectiveness in variance reduction, bias mitigation, and improved generalization within our framework. Finally, we validate our approach through experiments in assay scheduling, demonstrating significant improvements in resource utilization and decision quality over both traditional heuristic methods and value iteration (VI)-based approaches (see Supplementary Information for detailed comparisons).

2 Related Work

The optimization of decision-making under uncertainty is central to various fields, including drug discovery. Our work builds upon and contributes to research in Markov Decision Processes, Reinforcement Learning, Bayesian methods, and Experimental Design.

MDPs and Model-Based Reinforcement Learning : MDPs provide a mathematical framework for sequential decision-making (Puterman, 2014). While applied in drug discovery contexts like clinical trial optimization (Bennett & Hauser, 2013; Eghbali-Zarch et al., 2019; Abbas et al., 2007; Fard et al., 2018), their use in preclinical scheduling is limited by the difficulty of specifying transition probabilities ($P(s'|s,a)$) and rewards ($R(s,a)$). Model-Based Reinforcement Learning (MBRL) learns environmental models to improve planning (Sutton & Barto, 2018; Kaiser et al.; Moerland et al., 2023), with applications in molecule generation (Wang et al., 2021; Bengio et al., 2021; You et al., 2018; Zhou et al., 2019) and synthesis (Segler et al., 2018). However, learning accurate, explicit models for complex biological systems remains challenging. Kernel-based RL methods (Ormoneit & Sen, 2002; Kveton & Theocharous, 2012; Xu et al., 2007) use similarity functions, often for value function approximation or state generalization. In contrast, IB-MDP employs a similarity metric directly to create a *generative*, non-parametric transition model by sampling from historical data, rather than approximating values or learning explicit kernelized dynamics.

Bayesian RL, Optimization, and Similarity Metrics : Leveraging historical data under uncertainty is key. Bayesian Reinforcement Learning (BRL) typically maintains posteriors over model parameters

or value functions (Ghavamzadeh et al., 2015). Methods like Posterior Sampling RL (PSRL) (Osband et al., 2013; Agrawal & Jia, 2017) sample explicit MDP models from a posterior for planning. IB-MDP differs by avoiding explicit posteriors over MDP parameters; it captures uncertainty implicitly via similarity-weighted sampling from historical data, with weight updates acting as heuristic belief adjustments (Appendix H). Bayesian Optimization (BO), used for optimizing molecular properties (Griffiths & Hernández-Lobato, 2020; Gómez-Bombarelli et al., 2018), primarily targets single-step optimization, unlike IB-MDP's multi-step sequential planning with costs and state constraints. IB-MDP integrates similarity metrics (Bellet et al., 2013) directly into this sequential decision process via its sampling mechanism.

Ensemble Methods in Reinforcement Learning : Ensemble methods improve robustness and reliability (Dietterich, 2000; Zhou, 2012). In RL, they enhance exploration, generalization, and uncertainty estimation (Wiering & Van Hasselt, 2008; Osband et al., 2016; Lakshminarayanan et al., 2017). Our IB-MDP framework applies ensemble principles not as a novel method itself, but specifically to enhance decision robustness by aggregating policies from multiple runs, thereby mitigating variance introduced by its stochastic sampling-based model and MCTS planner.

Bayesian Experimental Design (BED), Implicit Models, and RL : Efficient data acquisition is studied in Bayesian Experimental Design (BED) (Chaloner & Verdinelli, 1995; Rainforth et al., 2024). A challenge arises with simulators or complex systems where likelihoods are intractable (implicit models). Recent work tackles **Implicit BED** using information-theoretic approaches (Kleinegesse & Gutmann, 2020; 2021) or policy learning (Ivanova et al., 2021). While sharing the implicit model philosophy, IB-MDP differs by embedding the problem within an RL (MDP) framework solved via multi-step MCTS planning, explicitly incorporating operational costs and state-based termination/feasibility constraints, using similarity-based sampling rather than focusing primarily on information-theoretic objectives. **RL for Sequential Experimental Design** (e.g., Blau et al., 2022) is related; IB-MDP contributes as a model-based RL variant using historical data for its implicit simulator.

Constrained MDPs and POMDPs : Our problem involves constraints on state properties (uncertainty $H(s)$, likelihood $L(s)$). This differs from typical Constrained MDPs (CMDPs) which often constrain cumulative action costs (Achiam et al., 2017); IB-MDP handles its constraints via rewards and termination conditions within MCTS. Furthermore, the task's information-gathering nature resembles a Partially Observable MDP (POMDP) (Kaelbling et al., 1998). As argued in Appendix H, IB-MDP's state representation and weight updates act as an implicit belief mechanism, justifying the dynamic transitions observed within MCTS as belief propagation.

Application Context: ADME Studies : ADME studies are crucial for establishing efficacy and safety profiles during drug discovery (Hoffman, 1998; Hoffman et al., 2004; Hughes et al., 2011). Decision-making here requires balancing information gain against resource limits. While RL has been applied to clinical trial optimization (Coronato et al., 2020; Escandell-Montero et al., 2014; Martín et al., 2020), its use for optimizing assay scheduling in drug discovery is less explored. IB-MDP addresses this gap with a flexible, scalable approach integrating historical and real-time data via its implicit, similarity-based model and ensemble planning, distinguishing it from traditional heuristics or explicit modeling techniques.

3 A Sequential Decision-Making Problem Statement

The sequential scheduling of experimental assays to define a drug's efficacy and safety profile is a significant challenge in drug discovery. Here, we focus on sequential assay decisions for central nervous system (CNS) drug candidates, where establishing whether a compound can cross the blood–brain barrier is critical. A key step involves identifying whether the compound is a substrate of efflux transporters such as P-glycoprotein (PgP) and Breast Cancer Resistance Protein (BCRP). While *in vitro* assays offer valuable early indicators of brain penetration potential, *in vivo* assays such as the measurement of unbound brain-to-plasma partition coefficient (k_{puu}) often provide a more definitive assessment. Consequently, the central goal in scheduling these assays is to maximize information on a compound's likelihood of crossing the blood–brain barrier while minimizing operational costs and resource constraints.

This inherently sequential decision-making problem, where choices influence future information states and costs, can be formulated as a multi-objective optimization problem under uncertainty. The primary objectives are to: (1) minimize the uncertainty in determining critical drug properties, particularly k_{puu}, and (2) minimize the total operational costs associated with conducting assays. These objectives must be balanced while adhering to resource constraints such as laboratory capacity and monetary costs. The challenge lies not in model uncertainty, but rather in efficiently reducing state uncertainty—the incomplete knowledge about the compound's hidden true properties (akin to a partially observable state)—through strategic selection and sequencing of experimental assays.

The detailed formulation of the IB-MDP algorithm and its implementation are provided in Appendix A.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Our experimental setup utilizes a dataset of 220 compounds, each characterized by both *in silico* predictions and physical properties. This dataset serves as the historical data $\mathcal{D}$ for IB-MDP. The *in silico* features include Quantitative Structure-Activity Relationship (QSAR) predictions, such as $\text{QSAR}_{\text{1uM\_PgP}}$, $\text{QSAR}_{\text{100nM\_BCRP}}$, and QSAR_{mrt}. In addition to these predictions, measured transporter activity data such as 100nM PgP, 1uM PgP, and 100nM BCRP are also available in the dataset. The financial and time costs associated with these transporter activities (actions) are estimated at \$400 per assay (7 days turnaround), while the target k_{puu} measurements incur \$4000 (21 days). These values highlight the resource constraints.

To generate Maximum Likelihood Action Sets Path (MLASP), we allow up to three parallel assays per decision step, enabling simultaneous operations. This setup potentially reduces state uncertainty more efficiently. Uncertainty $\mathcal{H}(s)$ is calculated as described in Section A.2.1, with a threshold $\epsilon = 10$ used for termination conditions (Section A.2.5) to capture meaningful differences.

We employ the IB-MDP algorithm (Appendix A), integrated with an MCTS-DPW solver. This solver runs for 20,000 iterations per decision step, with an exploration constant $c = 5.0$, and $N_e = 50$ ensembles, balancing exploration and exploitation.

The primary goal is to identify action sequences that achieve the greatest reduction in state uncertainty about the target assay (k_{puu}), while minimizing costs and satisfying likelihood constraints $\mathcal{L}(s) \geq \tau$, comparing favorably to simply performing all assays including the expensive target one.

Experimental Computing Resources: We performed IB-MDP simulations on an Apple M1 Pro chip (16GB RAM). For each compound simulation (across 50 ensembles), the estimated completion time was approximately 1 hour.

4.2 TRADITIONAL HEURISTIC DECISION RULES

As a practical baseline, the decision-making for brain penetration assays often relies on simple heuristic rules based on QSAR predictions and the final k_{puu} assay result: A compound is considered **promising** if: $\text{QSAR}_{\text{1uM\_PgP}} < 2$, $\text{QSAR}_{\text{100nM\_BCRP}} < 2$, and $0.5 \leq k_{\text{puu}} \leq 1$. A compound is considered **non-promising** if either: $\text{QSAR}_{\text{1uM\_PgP}} > 4$ or $\text{QSAR}_{\text{100nM\_BCRP}} > 4$, regardless of the k_{puu} value.

4.3 BENCHMARK POLICIES

To evaluate the effectiveness of IB-MDP against stronger baselines, we conducted a systematic benchmark study using synthetic datasets with known structures. We established a baseline using Value Iteration (VI) with exact theoretical uncertainty calculations (VI-Theo) and compared IB-MDP against both VI-Theo and a VI variant using similarity-based estimation (VI-Sim). Full results demonstrating IB-MDP's advantages are presented in the Supplementary Information (SI).

4.4 SELECTIVE CASE STUDY FOR COMPOUND SELECTION DECISION-MAKING

We tested the framework on the 220-compound dataset using three scenarios reflecting different QSAR conditions, demonstrating IB-MDP's flexibility and potential to identify promising compounds missed by traditional methods:

Baseline Confirmation: Tests compounds where QSARs are low (< 2) and k_{puu} is normal, validating standard heuristics.

Heuristic Challenge: Tests compounds with borderline/conflicting QSARs (at least one > 4), assessing IB-MDP's ability to interpret complex signals.

Opportunity Discovery: Evaluates compounds with high QSARs (> 4) but acceptable k_{puu}, seeking compounds overlooked by heuristics.

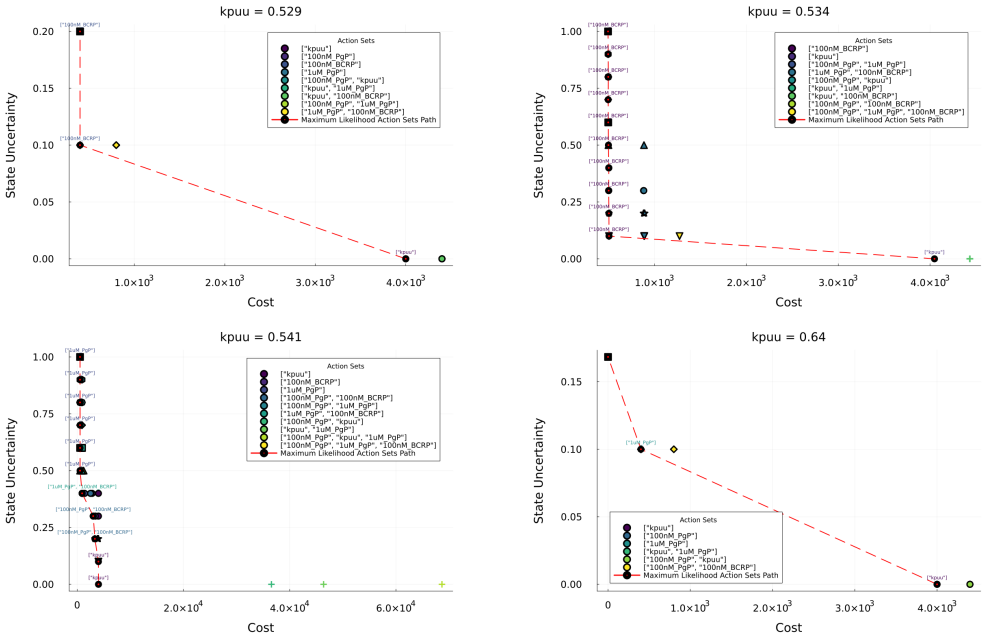


Figure 1: Monetary-prioritized IB-MDP results with MLASPs for 4 representative compounds, ordered by k_{puu} values to illustrate variations in QSAR prediction and recommended actions.
For k_{puu} = 0.53, $\text{QSAR}_{\text{1uM\_PgP}} = 5.0$, $\text{QSAR}_{\text{100nM\_BCRP}} = 9.6$, and $\text{QSAR}_{\text{mrt}} = 0.99$. The IB-MDP recommends action is [100nM_BCRP] (top left).
For k_{puu} = 0.53, $\text{QSAR}_{\text{1uM\_PgP}} = 0.903$, $\text{QSAR}_{\text{100nM\_BCRP}} = 8.5$, and $\text{QSAR}_{\text{mrt}} = 2.64$. The recommended action is [100nM_BCRP] (top right).
For k_{puu} = 0.54, $\text{QSAR}_{\text{1uM\_PgP}} = 1.68$, $\text{QSAR}_{\text{100nM\_BCRP}} = 1.3$, and $\text{QSAR}_{\text{mrt}} = 1.82$. The IB-MDP suggests actions are either [100nM_PgP, 100nM_BCRP] or [1uM_PgP, 100nM_BCRP] (bottom left).
For k_{puu} = 0.64, $\text{QSAR}_{\text{1uM\_PgP}} = 21.4$, $\text{QSAR}_{\text{100nM\_BCRP}} = 0.73$, and $\text{QSAR}_{\text{mrt}} = 1.2$. Recommended actions include [1uM_PgP], indicating a high probability of effectiveness under the given experimental conditions (bottom right).

4.5 EXPERIMENTAL RESULTS: COST COMPARISON BETWEEN CONVENTIONAL AND IB-MDP DECISIONS

The results of the IB-MDP exploration for representative cases (corresponding to the scenarios above and detailed in Table 1) are shown in Figure 1. In the baseline scenario, IB-MDP recommends sequences costing \$400-\$800, significantly less than the conventional approach of running all assays (\$5200 = $\$4000_{kpuu} + 3 \times \$400_{\text{transporter}}$).

In the heuristic challenge scenario, IB-MDP proposes efficient action sequences (e.g., single assay, \$400 cost), whereas traditional heuristics would discard potentially viable compounds. For the opportunity discovery scenario (high QSARs), IB-MDP successfully identifies cost-effective action sequences (e.g., [100nM_PgP, 1uM_PgP], cost \$800) that significantly reduce uncertainty, finding value where traditional rules fail.

Table 1: Comparison of Traditional Approach and IB-MDP Generated Costs for Selected Compounds

QSAR			Assay Values				Cost (\$) ($\times 100$)	
1uM PgP	100nM BCRP	mrt	kpuu	100nM PgP	1uM PgP	100nM BCRP	Trad.	IB-MDP
1.7	1.3	1.8	0.54	1.1	0.8	1.3	52	4-8
0.9	8.5	2.6	0.53	2.2	1.1	14.2	52	4
21.4	0.7	1.2	0.64	17.4	19.7	0.8	52	4-8
5.0	9.6	1.0	0.53	15.9	12.9	8.2	52	8

5 CONCLUSIONS

In this study, we introduced IB-MDP, a framework that improves decision-making under uncertainty in resource-constrained environments. Framed conceptually as an approximate POMDP (Appendix H), it dynamically integrates historical data via a similarity-based metric. Through implicit, heuristic belief updates (adaptive weight recalculation) and sampling processes within an MCTS planner, IB-MDP enables policies that maximize information gain, minimize costs, and meet key experimental objectives.

A notable advantage is its ability to reduce state uncertainty cost-effectively, often without relying solely on the most expensive assays, thereby improving efficiency and potentially accelerating decisions. Moreover, IB-MDP employs an ensemble approach (Dietterich, 2000; Lakshminarayanan et al., 2017) that aggregates multiple policy runs, reducing variance and enhancing robustness against the stochasticity of the model and planner. By generating a Maximum Likelihood Action Sets Path (MLASP) guided by likelihood constraints, it ensures more consistent decision quality.

Overall, IB-MDP outperforms traditional heuristic methods and simpler VI-based approaches (see SI), offering enhanced adaptability, precision, and resource optimization. This comprehensive, data-driven framework shows strong potential for streamlining sequential decision-making tasks in drug discovery and other fields requiring strategic resource management under uncertainty.

REFERENCES

Ismail Abbas, Joan Rovira, and Josep Casanovas. Clinical trial optimization: Monte carlo simulation markov model for planning clinical trials recruitment. In *Contemporary clinical trials*, volume 28, pp. 220–231. Elsevier, 2007.

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pp. 22–31. PMLR, 2017.

Shipra Agrawal and Randy Jia. Posterior sampling for reinforcement learning: worst-case regret bounds. *Advances in Neural Information Processing Systems*, 30, 2017.

Oguzhan Alagoz, Heather Hsu, Andrew J Schaefer, and Mark S Roberts. Markov decision processes: a tool for sequential decision making under uncertainty.

Aurélien Bellet, Amaury Habrard, and Marc Sebban. A survey on metric learning for feature vectors and structured data. *arXiv preprint arXiv:1306.6709*, 2013.

Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow network based generative models for non-iterative diverse candidate generation. *Advances in Neural Information Processing Systems*, 34:27381–27394, 2021.

Casey C Bennett and Kris Hauser. Artificial intelligence framework for simulating clinical decision-making: A markov decision process approach. *Artificial intelligence in medicine*, 57(1):9–19, 2013.

Tom Blau, Edwin V Bonilla, Iadine Chades, and Amir Dezfouli. Optimizing sequential experimental design with deep reinforcement learning. In *International Conference on Machine Learning*, pp. 2107–2128. PMLR, 2022.

Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43, 2012.

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical science*, pp. 273–304, 1995.

Antonio Coronato, Muddasar Naeem, Giuseppe De Pietro, and Giovanni Paragliola. Reinforcement learning for intelligent healthcare applications: A survey. *Artificial intelligence in medicine*, 109: 101964, 2020.

Adrien Couëtoux, Jean-Baptiste Hoock, Nataliya Sokolovska, Olivier Teytaud, and Nicolas Bonnard. Continuous upper confidence trees. In *Learning and Intelligent Optimization: 5th International Conference, LION 5, Rome, Italy, January 17-21, 2011. Selected Papers 5*, pp. 433–445. Springer, 2011.

Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pp. 1–15. Springer, 2000.

Maryam Eghbali-Zarch, Reza Tavakkoli-Moghaddam, Fatemeh Esfahanian, Amir Azaron, and Mohammad Mehdi Sepehri. A markov decision process for modeling adverse drug reactions in medication treatment of type 2 diabetes. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 233(8):793–811, 2019.

Pablo Escandell-Montero, Milena Chermisi, Jose M Martinez-Martinez, Juan Gomez-Sanchis, Carlo Barbieri, Emilio Soria-Olivas, Flavio Mari, Joan Vila-Francés, Andrea Stopper, Emanuele Gatti, et al. Optimization of anemia treatment in hemodialysis patients via reinforcement learning. *Artificial intelligence in medicine*, 62(1):47–60, 2014.

Mahdi M Fard, Sandor Szalma, Shashikant Vattikuti, and Gyan Bhanot. A bayesian markov decision process framework for optimal decision making in clinical trials. *IEEE Journal of Biomedical and Health Informatics*, 22(6):2061–2068, 2018.

Mohammad Ghavamzadeh, Shie Mannor, Joelle Pineau, and Aviv Tamar. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015.

Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

Ryan-Rhys Griffiths and José Miguel Hernández-Lobato. Constrained bayesian optimization for automatic chemical design using variational autoencoders. *Chemical science*, 11(2):577–586, 2020.

Amnon Hoffman. Pharmacodynamic aspects of sustained release preparations. *Advanced drug delivery reviews*, 33(3):185–199, 1998.

Amnon Hoffman, David Stepensky, Eran Lavy, Sara Eyal, Eytan Klausner, and Michael Friedman. Pharmacokinetic and pharmacodynamic aspects of gastroretentive dosage forms. *International journal of pharmaceutics*, 277(1-2):141–153, 2004.

John P Hughes, Simon Rees, Sonya B Kalindjian, and Roger K Philpott. Principles of early drug discovery. *British journal of pharmacology*, 162(6):1239–1249, 2011.

Desislava R Ivanova, Adam Foster, Simon Kleinegesse, Michael U Gutmann, and Tom Rainforth. Implicit deep adaptive design: Policy-based experimental design without likelihoods. In *Advances in Neural Information Processing Systems*, volume 34, pp. 25785–25798, 2021.

Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy H Campbell, Krzysztof Czechowski, Dumitru Erhan, Chelsea Finn, Patryk Kozakowski, Sergey Levine, et al. Model-based reinforcement learning for atari. *arXiv preprint arXiv:1903.00374*.

Steven Kleinegesse and Michael U Gutmann. Bayesian experimental design for implicit models by mutual information neural estimation. In *International conference on machine learning*, pp. 5316–5326. PMLR, 2020.

Steven Kleinegesse and Michael U Gutmann. Gradient-based bayesian experimental design for implicit models using mutual information lower bounds. *arXiv preprint arXiv:2105.04379*, 2021.

Branislav Kveton and Georgios Theocharous. Kernel-based reinforcement learning on representative states. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pp. 977–983, 2012.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Mercedes F Martín, Carmen Sánchez, and Eva Gómez. Markov decision processes for modeling and optimization of drug discovery. *Journal of Chemical Information and Modeling*, 60(5):2494–2506, 2020.

Thomas M Moerland, Joost Broekens, Aske Plaat, and Catholijn M Jonker. Model-based reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 16(1):1–118, 2023.

Dirk Ormoneit and Śaunak Sen. Kernel-based reinforcement learning. *Machine learning*, 49:161–178, 2002.

Ian Osband, Daniel Russo, and Benjamin Van Roy. More effective reinforcement learning via posterior sampling. In *Advances in neural information processing systems*, volume 26, 2013.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped DQN. *Advances in neural information processing systems*, 29, 2016.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024.

Marwin HS Segler, Mike Preuss, and Mark P Waller. Planning chemical syntheses with deep neural networks and symbolic ai. *Nature*, 555(7698):604–610, 2018.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Jike Wang, Chang-Yu Hsieh, Mingyang Wang, Xiaorui Wang, Zhenxing Wu, Dejun Jiang, Benben Liao, Xujun Zhang, Bo Yang, Qiaojun He, et al. Multi-constraint molecular generation based on conditional transformer, knowledge distillation and reinforcement learning. *Nature Machine Intelligence*, 3(10):914–922, 2021.

Marco A Wiering and Hado Van Hasselt. Ensemble algorithms in reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(4):930–936, 2008.

Xin Xu, Dewen Hu, and Xicheng Lu. Kernel-based least squares policy iteration for reinforcement learning. *IEEE transactions on neural networks*, 18(4):973–992, 2007.

Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems*, 31, 2018.

Zhenpeng Zhou, Steven Kearnes, Li Li, Richard N Zare, and Patrick Riley. Optimization of molecules via deep reinforcement learning. *Scientific reports*, 9(1):10752, 2019.

Zhi-Hua Zhou. *Ensemble methods: foundations and algorithms*. CRC press, 2012.

A IB-MDP ALGORITHM DETAILS

A.1 FRAMEWORK DESCRIPTION

The IB-MDP algorithm optimizes experimental scheduling in resource-constrained settings by leveraging historical data $\mathcal{D}$ through a distance-based similarity metric. Conceptually viewing the problem as a Partially Observable MDP (POMDP, see Appendix H), where the true state (e.g., full compound properties) is hidden, this framework aims to strategically select assays (actions) to minimize costs and maximize information gain (reduce uncertainty about the hidden state), particularly in high-dimensional decision spaces like preclinical PKPD studies.

The algorithm starts with a partially known initial state s (representing the observed features) and potential actions $\mathcal{A}$. As it explores, IB-MDP dynamically adjusts its strategy. By constructing an implicit generative model based on $\mathcal{D}$, IB-MDP avoids explicit parametrization of transition probabilities $P(s'|s,a)$. Instead, transitions in the observed state are simulated by sampling from historical data, weighted by similarity to the current observed state s. This implicitly models the effect of actions and subsequent observations on the agent's knowledge or belief about the hidden true state. This method reduces modeling complexity while retaining flexibility.

A key feature is its use of Monte Carlo Tree Search with Double Progressive Widening (MCTS-DPW) (Browne et al., 2012; Couëtoux et al., 2011), enabling efficient planning in large state(-action) spaces. MCTS effectively explores sequences of potential actions and their impact on the implicit belief state (represented jointly by the observed state s and similarity weights $W(s)$, see Appendix H). It balances exploration and exploitation to identify promising experimental sequences. The recalculation of similarity weights after each simulated action (the "Adaptive Weight Update" below, Section **??**) acts as a heuristic belief update within the MCTS simulations, adapting the model based on gathered information.

To further enhance robustness, IB-MDP integrates an ensemble method, aggregating policies from multiple independent runs to mitigate variance and bias from the sampling process and MCTS stochasticity, ensuring reliable decision recommendations.

A.2 IB-MDP FORMULATION

The Implicit Bayesian Markov Decision Process (IB-MDP) framework models the sequential decision-making process over observed states s using historical data $\mathcal{D}$ to guide actions. It is formally defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}_{\text{implicit}}, \mathcal{R}, \gamma \rangle$, where:

- **States ($\mathcal{S}$):** The space of observed states s, representing the agent's current knowledge (e.g., measured features derived from historical data and chosen actions).
- **Actions ($\mathcal{A}$):** The set of available assays or experiments that can be performed.
- **Implicit Transition Function ($\mathcal{T}_{\text{implicit}}$):** Defines probabilities $P(s'|s,a)$ for transitions between observed states based on similarity-weighted sampling from historical data $\mathcal{D}$, implicitly modeling state dynamics and observation gathering (detailed in Section A.2.3).
- **Reward Function ($\mathcal{R}$):** A function assigning value to state-action pairs, primarily penalizing costs and enforcing constraints (detailed in Section A.2.4).
- **Discount Factor (γ):** A factor $0 < \gamma \leq 1$ discounting future rewards.

A.2.1 KEY STATE-DEPENDENT METRICS: UNCERTAINTY AND LIKELIHOOD

The decision process is fundamentally guided by two key metrics derived from the current observed state s and its similarity to the historical dataset $\mathcal{D}$:

- **State Uncertainty $\mathcal{H}(s)$:** Quantifies the ambiguity about a target property k (e.g., k_{puu}) given the current observations. It is calculated as the weighted variance over the historical data:

$$\mathcal{H}(s) = \frac{\sum_{i=1}^{N} w_i(s) \left((D_i)_k - \bar{k}_w(s) \right)^2}{\sum_{j=1}^{N} w_j(s)} \tag{1}$$

where $(D_i)_k$ is the value of the target property k in historical data point D_i, $w_i(s)$ are the similarity weights (defined in Section A.2.3), $N = |\mathcal{D}|$, and $\bar{k}_w(s)$ is the similarity-weighted mean of the target property:

$$\bar{k}_w(s) = \frac{\sum_{i=1}^{N} w_i(s) \cdot (D_i)_k}{\sum_{j=1}^{N} w_j(s)}. \tag{2}$$

- **State Likelihood $\mathcal{L}(s)$:** Represents the confidence that the true target property k lies within desired bounds $[k_{\text{min}}, k_{\text{max}}]$. It is estimated as the weighted proportion of historical data consistent with these bounds:

$$\mathcal{L}(s) = \frac{\sum_{i=1}^{N} w_i(s) \cdot \mathbb{I}[(D_i)_k \in [k_{\text{min}}, k_{\text{max}}]]}{\sum_{j=1}^{N} w_j(s)}, \tag{3}$$

  where $\mathbb{I}[\cdot]$ is the indicator function.

A.2.2 Optimization Goal and Constraints

The objective is to find an optimal policy $\pi^* : \mathcal{S} \to \mathcal{A}$ that minimizes the expected total discounted cost, while ensuring the final state meets a predefined uncertainty threshold (ϵ) and intermediate states maintain sufficient likelihood (τ):

$$\pi^* = \min_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{T} \gamma^t R(s_t, \pi(s_t)) \right]$$

subject to the state-dependent constraints:

1. Terminal Uncertainty Constraint: $\mathcal{H}(s_T) \leq \epsilon$.
2. Intermediate Likelihood Constraint: $\mathcal{L}(s_t) \geq \tau, \quad \forall t = 0, \ldots, T-1$.

These constraints ensure the policy prioritizes informative and reliable decision paths, distinguishing this formulation from typical Constrained MDPs (CMDPs) which usually focus on cumulative action cost constraints (Achiam et al., 2017). These state-based constraints are managed within the MCTS planner via the reward structure and termination conditions.

A.2.3 The Implicit Dynamics Engine

The core of IB-MDP lies in its mechanism for simulating state transitions ($\mathcal{T}_{\text{implicit}}$) and implicitly updating beliefs, driven by similarity to historical data. This involves three key steps:

Similarity Weight Function

The relevance of each historical data point $D_i \in \mathcal{D}$ to the current observed state s is quantified by similarity weights $w_i(s)$. These are computed using a variance-normalized exponential kernel:

$$w_i(s) = \exp\left(-\lambda_w \cdot d(s, D_i)\right), \tag{4}$$

where λ_w is a global sensitivity parameter, and $d(s, D_i)$ is the distance calculated over the features k currently observed in state s:

$$d(s, D_i) = \sum_{k \in \text{ObservedFeatures}(s)} \lambda_k \cdot \frac{(s_k - (D_i)_k)^2}{\sigma_k^2}. \tag{5}$$

Here, s_k is the value of observed feature k in state s, $(D_i)_k$ is its value in historical point D_i, σ_k^2 is the empirical variance of feature k in $\mathcal{D}$, and λ_k are feature-specific scaling factors. Let $W(s) = \{w_i(s)\}_{i=1}^{N}$ be the vector of weights.

IMPLICIT TRANSITION MODELING VIA SAMPLING

The transition function $\mathcal{T}_{\text{implicit}}$ defines the probability $P(s'|s,a)$ of moving from the current observed state s to the next observed state s' upon taking action a. It is implicitly defined by leveraging the similarity weights $W(s)$ and the historical data $\mathcal{D}$:

$$P(s'|s,a) = \sum_{i=1}^{N} \frac{w_i(s)}{\sum_{j=1}^{N} w_j(s)} \cdot \mathbb{I}[s' = s \oplus \Delta s(a, D_i)], \tag{6}$$

where:

- $w_i(s)$ is the similarity weight for historical data point D_i relative to state s (Eq. equation 4).
- $\Delta s(a, D_i)$ is the change in observed state (e.g., newly revealed feature values) resulting from applying action a in the context of historical point D_i. This Δs serves as the implicit observation.
- $\oplus$ denotes the state update operation (e.g., augmenting s with information from Δs).
- $\mathbb{I}[\cdot]$ is the indicator function.
- $N = |\mathcal{D}|$.

This definition specifies the probability distribution over next states used implicitly by the MCTS planner. Operationally, a transition $s \to s'$ is simulated using the following generative process:

1. **Sample Historical Context:** Select a historical data point $D_{s_{\text{sampled}}}$ from $\mathcal{D}$ according to the distribution defined by the normalized similarity weights $W(s)/\sum_j w_j(s)$.
2. **Determine Action Outcome:** Identify the state change $\Delta s(a, D_{s_{\text{sampled}}})$ associated with action a under the conditions of $D_{s_{\text{sampled}}}$.
3. **Update Observed State:** Compute the resulting state $s' = s \oplus \Delta s(a, D_{s_{\text{sampled}}})$.

This procedure allows MCTS to sample next states according to the probabilities defined in Eq. equation 6 without explicitly calculating or storing the full transition matrix.

ADAPTIVE WEIGHT UPDATE (HEURISTIC BELIEF UPDATE)

Crucially, after simulating a transition to the new state s', the similarity weights are recalculated based on this updated state:

$$W'(s') = \{w_i(s')\}_{i=1}^{N} \quad \text{using Eq. equation 4 and equation 5 with } s'.$$

This recalculation dynamically adjusts the perceived relevance of historical data points in light of the new information gained (Δs). This mechanism acts as a *heuristic belief update*, refining the model's focus for subsequent simulation steps originating from s' within the MCTS rollout. (See Appendix H for a conceptual framing within the Partially Observable MDP context).

A.2.4 REWARD FUNCTION

The reward function $R(s,a)$ guides the policy towards cost-effective information gathering while strictly enforcing the operational constraints defined in Section A.2.2:

$$R(s,a) = \begin{cases} -\mathbf{c}(a) \cdot \boldsymbol{\lambda}, & \text{if } a \neq \text{eox and } \mathcal{L}(s) \geq \tau, \\ -M, & \text{if } a = \text{eox or } \mathcal{L}(s) < \tau. \end{cases} \tag{7}$$

Here, $\mathbf{c}(a)$ is the vector of costs (e.g., monetary, time) associated with action a, $\boldsymbol{\lambda}$ is a vector of trade-off parameters weighting different cost dimensions, eox (short for "end-of-experiment") represents the action to terminate the experiment sequence, and M is a large penalty value ensuring that constraint violations (checked before taking action a) or premature termination are strongly avoided. For simplicity, the cost $\mathbf{c}(a)$ is assumed to be independent of the state s in this formulation.

A.2.5 TERMINAL CONDITION

The sequential decision process, whether during MCTS simulation or actual experimental execution, terminates under any of the following conditions:

- The information goals are successfully achieved: The state s satisfies both $\mathcal{H}(s) \leq \epsilon$ and $\mathcal{L}(s) \geq \tau$.
- A predefined process limit (horizon H, e.g., maximum number of steps, total budget) is reached.
- The explicit termination action $a = \text{eox}$ is selected by the policy.

A.3 THE IB-MDP ALGORITHM IMPLEMENTATION

The IB-MDP framework supports sequential decision-making in resource-constrained settings, where the agent must determine an optimal sequence of experiments to maximize information gain while minimizing cost and uncertainty. To efficiently plan in such settings, the IB-MDP algorithm leverages a variant of Monte Carlo Tree Search (MCTS) called Double Progressive Widening (DPW).

A.3.1 SOLVING IB-MDP WITH MCTS-DPW

To solve the IB-MDP problem, we use Monte Carlo Tree Search with Double Progressive Widening (MCTS-DPW) (Browne et al., 2012; Couëtoux et al., 2011). MCTS-DPW is particularly well-suited for large state-action spaces where the number of feasible actions expands dynamically. It constructs a search tree through iterative simulations consisting of four key steps: **Selection**, **Expansion**, **Simulation**, and **Backpropagation**.

In the **Selection** step, the Upper Confidence Bound (UCB) policy is used to balance exploration and exploitation:

$$a = \arg\max_{a' \in \mathcal{A}(s)} \left(Q(s, a') + c\sqrt{\frac{\ln N_t(s)}{N_t(s, a')}} \right),$$

where $Q(s, a')$ is the estimated value of action a' in state s, $N_t(s)$ is the number of visits to state s, and $N_t(s, a')$ is the number of times action a' was taken from s.

Crucially, the **Simulation** step invokes the implicit transition model (Section A.2.1) and adaptive similarity-based weight update (Section A.2.3). This mechanism models belief evolution implicitly using weighted sampling from historical data. As more information is gathered, the transition behavior adapts, capturing belief refinement over time. While standard MCTS convergence guarantees may not strictly hold under such non-stationarity, MCTS remains an effective heuristic for planning under implicit partial observability.

A.3.2 PARETO FRONT GENERATION

Each individual IB-MDP run produces a terminal state s_T by executing its corresponding policy π_j^*. After N_e ensemble runs, we collect the terminal outcomes $\{s_T^{(j)}\}$ and generate the empirical Pareto front across these runs.

The Pareto front is constructed by evaluating trade-offs between terminal cost $\mathcal{C}(s_T^{(j)})$ and uncertainty $\mathcal{H}(s_T^{(j)})$ for each trajectory. A state s'_T dominates another state s_T if $\mathcal{H}(s'_T) < \mathcal{H}(s_T)$ and $\mathcal{C}(s'_T) < \mathcal{C}(s_T)$. The resulting Pareto front thus includes only the non-dominated terminal outcomes from the ensemble, representing the best observed trade-offs between minimizing cost and reducing uncertainty.

This procedure allows us to visualize and evaluate the effectiveness of the IB-MDP planner across runs, revealing the diversity and robustness of achievable decision paths under different stochastic realizations.

A.4 ENSEMBLE METHOD FOR IB-MDP

The ensemble method enhances robustness by averaging over multiple independent runs of the IB-MDP planner, each affected by stochasticity from historical data sampling and MCTS exploration. Running the algorithm N_e times produces a set of candidate policies $\{\pi_j^*\}$ and outcome distributions. Aggregating across this ensemble reduces variance, mitigates sampling noise, and improves confidence in the final decision recommendation. This approach follows ensemble learning principles described in Dietterich (2000) and ensures greater stability in high-stakes experimental design tasks.

Advantages of Ensemble IB-MDP: The ensemble IB-MDP methodology provides several advantages:

- **Improved Robustness:** By aggregating results from multiple simulations, the ensemble approach reduces the impact of variability and randomness in individual runs, enhancing the stability of decision outcomes against the stochastic elements of the algorithm.
- **Bias Reduction:** Exploring diverse decision trajectories across the ensemble potentially minimizes inference bias that might arise from a single MCTS search getting stuck in a suboptimal region of the policy space. This yields more accurate estimates of effective action sequences.
- **Predictive Power & Action Prioritization:** The ensemble method helps identify frequently recommended actions or action sequences across multiple runs. The aggregation of these results informs the construction of a Maximum Likelihood Action Sets Path (MLASP), providing a robust, prioritized sequence of recommended actions guided by the collective results of the ensemble.

A.4.1 MAXIMUM LIKELIHOOD ACTION SETS PATH (MLASP)

The MLASP is a key outcome of the ensemble approach, providing a practical recommendation for decision-makers. It is constructed by identifying the most frequently occurring optimal action set A_u^* recommended by the ensemble policies at different stages, often characterized by the achieved uncertainty level u. Specifically, for representative uncertainty levels u, the most frequent action set is chosen:$\mathrm{A}_u^* = \arg\max_A \sum_{j=1}^{N_e} \mathbb{I}(A \text{ is part of policy } \pi_j^* \text{ at uncertainty } u)$. where $\mathbb{I}(\cdot)$ is the indicator function, counting how often action set A was part of the j-th policy π_j^* when the system state had uncertainty approximately equal to u. N_e is the total number of ensemble runs. By connecting the sequence of A_u^* across decreasing uncertainty levels (from initial state uncertainty down to the target ϵ), we form the MLASP path. This path represents a robust, cost-vs-uncertainty trade-off policy derived from the ensemble. Figure A.4.1 shows exemplary MLASP paths generated for the same compound but using different likelihood thresholds (τ), demonstrating how this constraint influences the recommended sequence of actions. Figure A.4.1 illustrates the action frequency distribution at a specific point used in the voting process to determine one step (A_u^*) along the MLASP. This aggregation leverages the ensemble to converge on reliable decisions despite simulation variability inherent in the IB-MDP process.

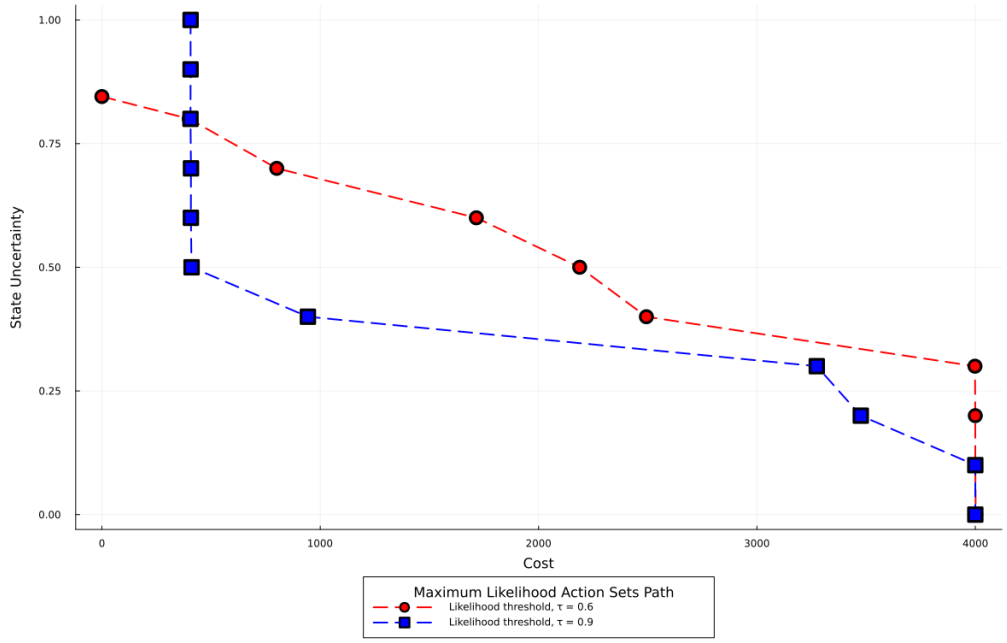
A.5 ALGORITHM

the detailed and complete description of the algorithm, see Algorithm 1.

B BENCHMARK TEST OVERVIEW

To evaluate the effectiveness of similarity-based approaches (i.e., IB-MDP) in experimental design, we conducted a systematic benchmark study. First, we generated a synthetic dataset with known uncertainty structures and feature relationships. This dataset served as our testing ground for comparing different decision-making approaches.

Using this dataset, we established a baseline policy through Value Iteration (VI) with exact theoretical calculations of uncertainty reduction. This baseline represents the optimal policy under perfect information about uncertainty dynamics. We then benchmarked two similarity-based approaches against this theoretical baseline: 1. Similarity-based Value Iteration (VI-Sim): This approach maintains the



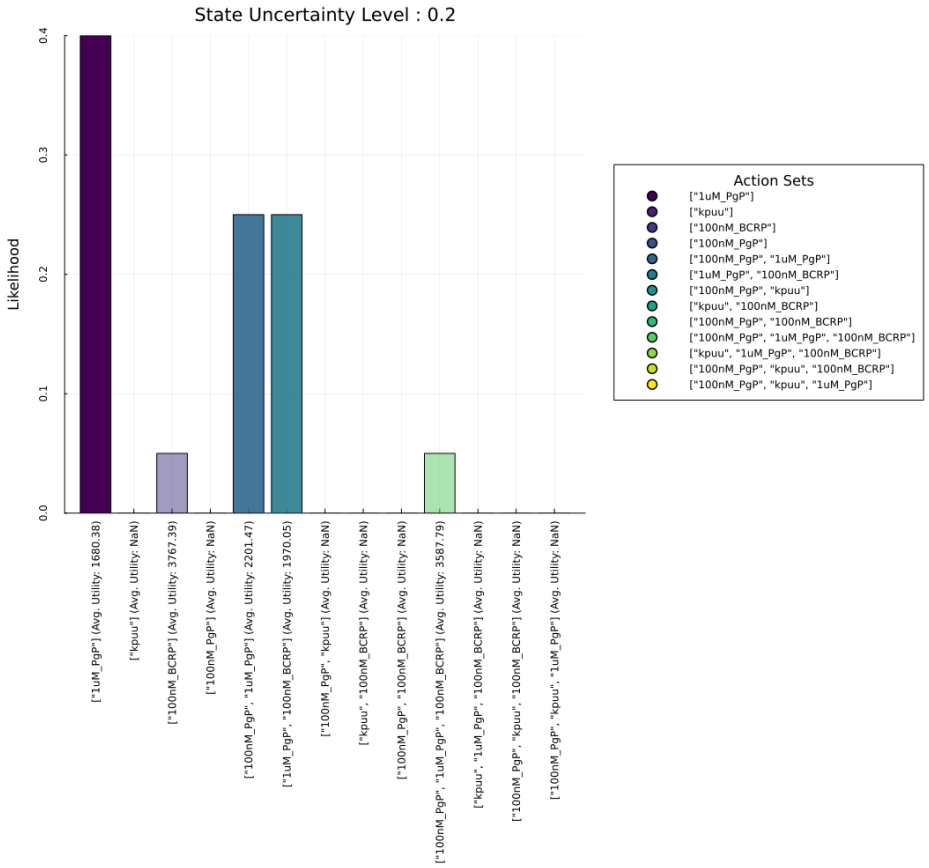
Algorithm 1 Ensemble IB-MDP Algorithm (Appendix)

Require: Initial state s_0, historical data $\mathcal{D}$, similarity function W, Bayesian update function β, horizon H, number of iterations n_{itr}, number of ensemble runs N_e, number of times a particular state 's' has been visited N_t

Ensure: Pareto front of state uncertainty vs. expected utility costs

Initialize an array $\mathcal{P}$ to store Pareto fronts
for $j = 1$ to N_e **do**
  Initialize MCTS-DPW tree with root node representing s_0
  for $i = 1$ to n_{itr} **do**
    $s \leftarrow s_0$
    while not terminal and within horizon H **do**
      Select action a using UCB policy: $a = \arg\max_{a' \in \mathcal{A}(s)} Q(s, a') + c\sqrt{\frac{\ln N_t(s)}{N_t(s,a')}}$
      Simulate next state s' using Bayesian update via sampling: $s' = \beta(s, \mathcal{D}, a)$
      Update similarity weights W based on new state s'
      Update tree with s' and reward $R(s, a)$
      $s \leftarrow s'$
    end while
    Backpropagate rewards and update Q values along the path
  end for
  $\pi_j^* \leftarrow$ Extract optimal policy from tree
  $\mathcal{P}_j \leftarrow$ Compute Pareto front from π_j^*
  Append $\mathcal{P}_j$ to $\mathcal{P}$
end for
for each uncertainty level u **do**
  $A_u^* = \arg\max_A \sum_{j=1}^{N_e} \mathbb{I}(A \in \mathcal{P}_j(u))$
end for
Construct Maximum Likelihood Action Sets Path (MLASP) from A_u^*
return MLASP

VI framework but replaces exact uncertainty calculations with similarity-based estimations using nearest neighbors. 2. IB-MDP: This approach combines similarity-based uncertainty estimation with Monte Carlo Tree Search to generate an ensemble of policies, providing both primary and alternative action recommendations.



This benchmark framework allows us to assess how well similarity-based methods can approximate the theoretically optimal policy when exact uncertainty calculations are unavailable or computationally prohibitive in real experimental scenarios.

C Synthetic Data Generation Process

The synthetic dataset is designed to simulate a chemical compound measurement system with multiple features and a target variable. The generation process follows a structured approach to create realistic correlations between features and the target.

C.0.1 Feature Generation

For a system with n features, each feature x_i is generated from a truncated normal distribution:

$$\begin{aligned} x_i &\sim \mathcal{TN}(\mu_i, \sigma_i, 0, 2\mu_i) \quad \text{for } i \in \{1, \ldots, n\} \\ \mu_i &= 50 \cdot \frac{i}{n} \\ \sigma_i &= 0.3\mu_i \end{aligned} \tag{8}$$

where $\mathcal{TN}$ represents a truncated normal distribution with lower bound 0 and upper bound $2\mu_i$. Here, μ_i represents the mean of feature i, and σ_i represents its standard deviation. The mean increases linearly with the feature index, while the standard deviation is set to 30% of the mean to maintain consistent relative variability across features. This ensures that:

- Features have different scales but consistent relative variability

- All feature values remain positive
- The variance increases proportionally with the mean

C.0.2 TARGET VARIABLE GENERATION

The target variable y is generated as a linear combination of the features with additive noise:

$$y = \sum_{i=1}^{n} \beta_i x_i + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2) \tag{9}$$

where:

- $\boldsymbol{\beta} = [\beta_1, ..., \beta_n]$ are the target coefficients
- β_i decreases with i to simulate varying feature importance
- σ_ϵ^2 represents measurement noise

In the data simulation, we choose to setup the dataset consisting of 200 compounds, each characterized by these six features and their corresponding target values.

- $\boldsymbol{\beta} = [0.3, 0.25, 0.2, 0.15, 0.07, 0.03]$ are the default target coefficients
- $\epsilon \sim \mathcal{TN}(0, 5, -10, 10)$ represents measurement noise

The measurement noise in our simulation is modeled using a truncated normal distribution:

$$\epsilon \sim \mathcal{TN}(\mu_\epsilon, \sigma_\epsilon, a, b) \tag{10}$$

with parameters:

$$\begin{aligned} \mu_\epsilon &= 0 \quad \text{(zero mean)} \\ \sigma_\epsilon &= 5 \quad \text{(standard deviation)} \\ a &= -10 \quad \text{(lower bound)} \\ b &= 10 \quad \text{(upper bound)} \end{aligned} \tag{11}$$

The truncation ensures that the noise remains within reasonable bounds relative to the target values, while the zero mean preserves the expected value of the target variable.

D BASELINE POLICY: VALUE ITERATION WITH EXACT UNCERTAINTY REDUCTION

Value Iteration (VI) provides a systematic approach to find optimal policies in sequential decision-making problems. For our experimental design task, we implement VI with exact uncertainty calculations to establish a baseline policy.

D.1 VALUE ITERATION FRAMEWORK

The general form of value iteration updates the value function $V(s)$ for each state s through iterative application of the Bellman equation:

$$V_{t+1}(s) = \max_{a \in A} \left\{ R(s, a) + \gamma \sum_{s' \in S} P(s'|s, a) V_t(s') \right\} \tag{12}$$

where $R(s, a)$ is the immediate reward, γ is the discount factor, and $P(s'|s, a)$ is the transition probability.

D.2 VALUE FUNCTION FOR EXPERIMENTAL DESIGN

In our experimental design context, we define:

- State $s = \mathcal{M}$: the set of measured features
- Action space $A(\mathcal{M}) = \mathcal{P}(\{1, \ldots, n\} \setminus \mathcal{M})$: the power set of unmeasured features
- Action $a \in A(\mathcal{M})$: a subset of features to measure next
- Reward $R(\mathcal{M}, a) = \frac{\Delta\sigma_a^2}{c_a}$: uncertainty reduction per unit cost

Since our measurement process is deterministic (measuring a feature always yields its true value), the transition probability $P(s'|s, a)$ simplifies to:

$$P(s'|s, a) = \begin{cases} 1 & \text{if } s' = s \cup a \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

This deterministic transition leads to our simplified value function:

$$V(\mathcal{M}) = \max_{a \in A(\mathcal{M})} \left\{ \frac{\Delta\sigma_a^2}{c_a} + \gamma V(\mathcal{M} \cup a) \right\} \tag{14}$$

D.3 THEORETICAL UNCERTAINTY CALCULATION

Given the linear relationship between features and target, we can calculate the exact uncertainty reduction for any sequence of measurements.

D.3.1 PRIOR UNCERTAINTY

The prior variance of the target variable, before any measurements, is:

$$\sigma_{prior}^2 = \sum_{i=1}^{n} \beta_i^2 \sigma_i^2 + \sigma_\epsilon^2 \tag{15}$$

where:

- β_i are the known target coefficients $[0.3, 0.25, 0.2, 0.15, 0.07, 0.03]$
- σ_i is the standard deviation of feature i: $\sigma_i = 0.3\mu_i$
- σ_ϵ is the noise standard deviation: $\sigma_\epsilon = 5$

D.3.2 CONDITIONAL UNCERTAINTY

After measuring a subset of features $\mathcal{M}$, the conditional variance is:

$$\sigma_{\text{conditional}}^2 = \sum_{i \notin \mathcal{M}} \beta_i^2 \sigma_i^2 + \sigma_\epsilon^2 \tag{16}$$

The uncertainty reduction from measuring feature set a is:

$$\Delta\sigma_a^2 = \sum_{k \in a} \beta_k^2 \sigma_k^2 \tag{17}$$

D.4 IMPLEMENTATION DETAILS

The value iteration algorithm is implemented with the following specifications:

- Discount factor $\gamma = 0.95$
- Convergence threshold $\epsilon = 10^{-6}$

- Maximum iterations: 1000
- Action space $A(\mathcal{M})$ includes all possible combinations of unmeasured features (full power set)
- Feature costs increase with index:

$$c_k = \begin{cases} 1.0 & \text{for } k = 1 \\ 1.2 & \text{for } k = 2 \\ 1.5 & \text{for } k = 3 \\ 1.8 & \text{for } k = 4 \\ 2.0 & \text{for } k = 5 \\ 2.2 & \text{for } k = 6 \\ 10.0 & \text{for target measurement} \end{cases} \tag{18}$$

For any action $a \in A(\mathcal{M})$ that includes multiple features, the total cost is the sum of individual feature costs:

$$c_a = \sum_{k \in a} c_k \tag{19}$$

The resulting policy provides the optimal feature selection strategy when exact uncertainty dynamics are known, serving as our baseline for comparing similarity-based approaches.

E Similarity-Based Policy Generation

To evaluate alternative approaches to the baseline Value Iteration policy with exact calculations, we investigate two similarity-based methods for generating experimental design policies. These methods are particularly relevant when exact uncertainty calculations are unavailable in real experimental scenarios.

E.1 Value Iteration with Similarity-Based Estimation

Our first approach maintains the Value Iteration framework while replacing exact uncertainty calculations with similarity-based estimations:

$$V(\mathcal{M}) = \max_{a \in A(\mathcal{M})} \left\{ \frac{\Delta \hat{\sigma}_a^2}{c_a} + \gamma V(\mathcal{M} \cup a) \right\} \tag{20}$$

where $\Delta \hat{\sigma}_a^2$ represents the estimated uncertainty reduction based on similarity metrics between experimental states. The estimated uncertainty reduction for action a is then:

$$\Delta \hat{\sigma}_a^2 = \hat{\sigma}^2_{conditional}(\mathcal{M}) - \hat{\sigma}^2_{conditional}(\mathcal{M} \cup a) \tag{21}$$

E.2 Similarity-Based Uncertainty Estimation

E.2.1 Method Description

The similarity-based approach estimates conditional variance using:

$$\hat{\sigma}^2_{conditional}(\mathcal{M}) = \sum_{j=1}^{N} w_j (y_j - \bar{y}_w)^2 \tag{22}$$

where:

- $w_j = \frac{\exp(-d(\mathbf{x}_j, \mathbf{x}))}{\sum_k \exp(-d(\mathbf{x}_k, \mathbf{x}))}$ are similarity weights
- $d(\mathbf{x}_j, \mathbf{x})$ is a distance metric over measured features $\mathcal{M}$
- $\bar{y}_w = \sum_{j=1}^{N} w_j y_j$ is the weighted mean
- N is the number of historical data points

E.3 IB-MDP POLICY GENERATION

Our second approach, IB-MDP, generates experimental policies using a combination of similarity-based uncertainty estimation and Monte Carlo Tree Search. The ensemble IB-MDP method:

- Generates an ensemble of potential policies
- Provides ranked action recommendations
- Incorporates exploration-exploitation trade-offs
- Adapts to varying experimental conditions

F CONVERGENCE ANALYSIS OF SIMILARITY-BASED VALUE ITERATION

F.1 THEOREM

For the synthetic data model with independent features, as the number of historical samples $N \to \infty$, the similarity-based variance estimate converges in probability to the theoretical variance:

$$\hat{\text{Var}}(y|\mathbf{x}_{\mathcal{M}}) = \hat{\sigma}^2_{\text{conditional}}(\mathcal{M}) \xrightarrow{p} \sum_{i \in \mathcal{U}} \beta_i^2 \sigma_i^2 + \sigma_\epsilon^2 \tag{23}$$

F.2 PROOF

1) For a fixed query point $\mathbf{x}$, the similarity weights w_j form a probability distribution:

$$\begin{aligned} w_j &= \frac{\exp(-d(\mathbf{x}_j, \mathbf{x}))}{\sum_{k=1}^N \exp(-d(\mathbf{x}_k, \mathbf{x}))} \\ \sum_{j=1}^N w_j &= 1, \quad w_j \geq 0 \quad \forall j \end{aligned} \tag{24}$$

where $d(\mathbf{x}_j, \mathbf{x})$ is the distance metric over measured features.

2) Due to the independence of features in our model:

$$P(x_i|\mathbf{x}_{\mathcal{M}}) = P(x_i) \quad \forall i \in \mathcal{U} \tag{25}$$

where $x_i \sim \mathcal{TN}(\mu_i, \sigma_i^2, 0, 2\mu_i)$ for unmeasured features.

3) The weighted variance estimator can be decomposed:

$$\begin{aligned} \hat{\text{Var}}(y|\mathbf{x}_{\mathcal{M}}) &= \sum_{j=1}^N w_j (y_j - \bar{y}_w)^2 \\ &= \sum_{j=1}^N w_j (\sum_{i \in \mathcal{U}} \beta_i x_{ji} + \epsilon_j - (\sum_{i \in \mathcal{U}} \beta_i \bar{x}_i + \bar{\epsilon}))^2 \end{aligned} \tag{26}$$

where $\bar{x}_i = \sum_{j=1}^N w_j x_{ji}$ and $\bar{\epsilon} = \sum_{j=1}^N w_j \epsilon_j$.

4) By the strong law of large numbers, as $N \to \infty$:

$$\begin{aligned} \bar{x}_i &\xrightarrow{a.s.} \mathbb{E}[x_i] = \mu_i \quad \forall i \in \mathcal{U} \\ \bar{\epsilon} &\xrightarrow{a.s.} \mathbb{E}[\epsilon] = 0 \end{aligned} \tag{27}$$

5) Therefore, by the continuous mapping theorem:

$$\begin{aligned}\hat{\text{Var}}(y|\mathbf{x}_{\mathcal{M}}) &\xrightarrow{p} \mathbb{E}[(y - \mathbb{E}[y|\mathbf{x}_{\mathcal{M}}])^2|\mathbf{x}_{\mathcal{M}}] \\ &= \text{Var}(\sum_{i\in\mathcal{U}} \beta_i x_i + \epsilon|\mathbf{x}_{\mathcal{M}}) \\ &= \sum_{i\in\mathcal{U}} \text{Var}(\beta_i x_i|\mathbf{x}_{\mathcal{M}}) + \text{Var}(\epsilon|\mathbf{x}_{\mathcal{M}}) \\ &= \sum_{i\in\mathcal{U}} \beta_i^2\sigma_i^2 + \sigma_\epsilon^2\end{aligned} \tag{28}$$

The final equality follows from:

- Independence of features: $\text{Var}(x_i|\mathbf{x}_{\mathcal{M}}) = \text{Var}(x_i) = \sigma_i^2$
- Independence of noise: $\text{Var}(\epsilon|\mathbf{x}_{\mathcal{M}}) = \sigma_\epsilon^2$
- No covariance terms due to independence

The convergence result has significant implications for linear relationships, where the similarity-based approach provides consistent estimates of uncertainty reduction. Under the assumptions of linear relationships and independent features, this method yields consistent estimates while remaining unbiased, although it may exhibit higher variance in finite samples. However, it is important to note that the approach is limited to linear relationships and may not accurately capture non-linear relationships or interactions. The approach's convergence rate is influenced by various factors, including the dimension of the measured feature space, the density of available data points, and the choice of similarity metric and bandwidth parameters. While both the similarity-based approach and theoretical calculations yield equivalent results in the limit for linear models, they differ in practical application. Theoretical calculations provide exact results for synthetic models with known linear structure, making them computationally efficient. The similarity-based approach, while still restricted to linear relationships, has the advantage of not requiring explicit knowledge of the coefficients and scales computationally with data size. However, for real-world scenarios with potential non-linear relationships, both methods may be inadequate, and more sophisticated uncertainty estimation techniques should be considered.

G Benchmark Test Results

To evaluate different approaches for experimental design policy generation, we conducted extensive benchmark tests comparing a theoretical baseline against two alternatives: similarity-based Value Iteration and Implicit Bayesian MDP (IB-MDP). Each approach was tested over 100 iterations to assess their consistency and reliability.

G.1 Test Methodology

For each iteration, we follow these steps:

1. Data Generation

- Generate 200 compounds using our synthetic data model
- Features follow truncated normal distributions with known parameters
- Target variable calculated using fixed coefficients with optional noise

2. Policy Generation

- VI Theoretical (VI Theo): Baseline policy using exact uncertainty calculations and known coefficients
- VI Similarity (VI Sim): Policy using similarity-based uncertainty estimation without knowledge of coefficients

- Implicit Bayesian MDP (IB-MDP): Policy using ensemble-based approach with Monte Carlo Tree seaerch algorithm

3. Feature Recommendations

- VI Theo generates a single optimal feature recommendation
- VI Sim generates a single feature recommendation based on similarity metrics
- IB-MDP generates primary (Top1) and alternative (Top2) feature recommendations

G.2 Comparison Metrics

We evaluate three alignment metrics:

- T1 Match: Binary indicator (1/0) if IB-MDP's primary recommendation contains the theoretical optimal feature
- T2 Match: Binary indicator (1/0) if IB-MDP's alternative recommendation contains the theoretical optimal feature
- Sim Match: Binary indicator (1/0) if VI Similarity's recommendation matches the theoretical choice

G.3 Results Table

The Table 2 presents iteration-by-iteration results where:

- Iter: Iteration number (1-100)
- VI Theo: Feature recommended by theoretical Value Iteration
- IB-MDP Top1 Features: Set of features in primary recommendation
- IB-MDP Top2 Features: Set of features in alternative recommendation
- VI Sim: Feature recommended by similarity-based Value Iteration
- Match columns: Binary indicators for alignment with theoretical recommendation

Table 2: Comparison of VI Theoretical vs IB-MDP and VI Similarity Approaches

Iter	VI Theo	IB-MDP Top1 Features	T1 Match	IB-MDP Top2 Features	T2 Match	VI Sim	Sim Match
1	5	{3, 4}	0	{3, 5}	1	3	0
2	5	{3, 4}	0	{3, 5, 6}	1	3	0
3	3	{3, 4}	1	{3, 5}	1	5	0
4	5	{3, 4}	0	{3, 5}	1	3	0
5	3	{3, 4}	1	{3, 5}	1	3	1
6	4	{3, 4}	1	{3, 5}	0	3	0
7	3	{3, 4}	1	{3, 5}	1	3	1
8	5	{3, 4}	0	{3, 5, 6}	1	3	0
9	3	{3, 4}	1	{3, 5}	1	3	1
10	4	{3, 4}	1	{3, 5, 6}	0	6	0
11	4	{3, 4}	1	{3, 5}	0	4	1
12	4	{3, 4}	1	{3, 5}	0	4	1
13	4	{3, 4}	1	{3, 5}	0	6	0
14	4	{3, 4}	1	{3, 5}	0	6	0
15	5	{3, 4}	0	{3, 5}	1	3	0
16	5	{3, 4}	0	{3, 5}	1	3	0
17	4	{3, 4}	1	{3, 5}	0	4	1
18	4	{3, 4}	1	{3, 5, 6}	0	3	0
19	5	{3, 4}	0	{3, 5}	1	3	0

Table 2: Comparison of VI Theoretical vs IB-MDP and VI Similarity Approaches

Iter	VI Theo	IB-MDP Top1 Features	T1 Match	IB-MDP Top2 Features	T2 Match	VI Sim	Sim Match
20	5	{3, 4}	0	{3, 5}	1	3	0
21	5	{3, 4}	0	{3, 5}	1	3	0
22	4	{3, 4}	1	{3, 5}	0	4	1
23	5	{3, 4}	0	{3, 5}	1	3	0
24	4	{3, 4}	1	{3, 5}	0	6	0
25	5	{3, 4}	0	{3, 6}	0	5	1
26	5	{3, 4}	0	{3, 5}	1	3	0
27	5	{3, 4}	0	{3, 5}	1	3	0
28	4	{3, 4}	1	{3, 5}	0	3	0
29	4	{3, 4}	1	{3, 5}	0	3	0
30	5	{3, 4}	0	{3, 5}	1	3	0
31	5	{3, 4}	0	{3, 5, 6}	1	3	0
32	4	{3, 4}	1	{3, 5}	0	6	0
33	5	{3, 4}	0	{3, 5}	1	5	1
34	4	{3, 4}	1	{3, 5}	0	4	1
35	4	{3, 4}	1	{3, 5}	0	4	1
36	5	{3, 4}	0	{3, 5}	1	3	0
37	5	{3, 4}	0	{3, 5}	1	3	0
38	5	{3, 4}	0	{3, 5}	1	3	0
39	5	{3, 4}	0	{3, 5}	1	3	0
40	3	{3, 4}	1	{3, 5}	1	3	1
41	4	{3, 4}	1	{3, 5}	0	4	1
42	4	{3, 4, 5}	1	{3, 4, 5}	1	6	0
43	3	{3, 4}	1	{3, 5}	1	3	1
44	5	{3, 4}	0	{3, 5, 6}	1	3	0
45	5	{3, 4}	0	{3, 5}	1	3	0
46	5	{3, 4}	0	{3, 5}	1	3	0
47	3	{3, 4}	1	{3, 5}	1	3	1
48	5	{3, 4}	0	{3, 5}	1	5	1
49	4	{3, 4}	1	{3, 5}	0	3	0
50	5	{3, 4}	0	{3, 5}	1	3	0
51	4	{3, 4}	1	{3, 5, 6}	0	4	1
52	5	{3, 4}	0	{3, 5}	1	3	0
53	5	{3, 4}	0	{3, 5}	1	5	1
54	5	{3, 4}	0	{3, 5}	1	3	0
55	4	{3, 4}	1	{3, 5}	0	3	0
56	5	{3, 4}	0	{3, 5}	1	3	0
57	4	{3, 4}	1	{3, 4, 5}	1	3	0
58	3	{3, 4}	1	{3, 5}	1	3	1
59	4	{3, 4}	1	{3, 5}	0	6	0
60	3	{3, 4}	1	{3, 4, 5}	1	3	1
61	3	{3, 4}	1	{3, 5, 6}	1	3	1
62	6	{3, 4}	0	{3, 5}	0	3	0
63	5	{3, 4}	0	{3, 5}	1	5	1
64	4	{3, 4}	1	{3, 5}	0	3	0
65	5	{3, 4}	0	{3, 5}	1	3	0
66	5	{3, 4}	0	{3, 5}	1	3	0
67	5	{3, 4}	0	{3, 5}	1	3	0
68	4	{3, 4}	1	{3, 5}	0	6	0
69	6	{3, 4}	0	{3, 5}	0	6	1
70	4	{3, 4}	1	{3, 5}	0	4	1
71	5	{3, 4}	0	{3, 5}	1	3	0
72	5	{3, 4}	0	{3, 5}	1	3	0
73	4	{3, 4}	1	{3, 5}	0	6	0

Table 2: Comparison of VI Theoretical vs IB-MDP and VI Similarity Approaches

Iter	VI Theo	IB-MDP Top1 Features	T1 Match	IB-MDP Top2 Features	T2 Match	VI Sim	Sim Match
74	5	{3, 4}	0	{3, 5}	1	3	0
75	5	{3, 4}	0	{3, 5}	1	5	1
76	5	{3, 4}	0	{3, 5}	1	5	1
77	5	{3, 4}	0	{3, 5}	1	3	0
78	4	{3, 4}	1	{3, 5}	0	4	1
79	5	{3, 4}	0	{3, 5}	1	3	0
80	5	{3, 4}	0	{3, 5}	1	5	1
81	5	{3, 4}	0	{3, 5}	1	3	0
82	5	{3, 4}	0	{3, 5}	1	3	0
83	4	{3, 4}	1	{3, 5}	0	4	1
84	5	{3, 4}	0	{3, 5}	1	3	0
85	5	{3, 4}	0	{3, 5}	1	3	0
86	5	{3, 4}	0	{3, 5}	1	3	0
87	4	{3, 4}	1	{3, 5, 6}	0	3	0
88	5	{3, 4}	0	{3, 5}	1	3	0
89	5	{3, 4}	0	{3, 5}	1	3	0
90	5	{3, 4}	0	{3, 5}	1	3	0
91	3	{3, 4}	1	{3, 5}	1	3	1
92	4	{3, 4}	1	{3, 5}	0	4	1
93	3	{3, 4}	1	{3, 5}	1	3	1
94	3	{3, 4}	1	{3, 5}	1	3	1
95	4	{3, 4}	1	{3, 5, 6}	0	3	0
96	5	{3, 4}	0	{3, 5}	1	5	1
97	5	{3, 4}	0	{3, 5}	1	3	0
98	4	{3, 4}	1	{3, 5}	0	4	1
99	3	{3, 4}	1	{3, 5}	1	3	1
100	4	{3, 4}	1	{3, 5}	0	6	0

G.4 Benchmark Results Analysis

Table 3: Policy Alignment with Theoretical Baseline

Method	Matches	Match Rate (%)
IB-MDP Top 1	47	47.0
IB-MDP Top 2	66	66.0
VI Similarity	36	36.0

The benchmark results, summarized in Table 3, reveal systematic differences between deterministic VI-based policies and ensemble-based IB-MDP policies. Over 100 iterations, IB-MDP's primary recommendations achieve 47

Value Iteration approaches, both theoretical and similarity-based, produce deterministic policies through the optimization of $\pi^*(s) = \arg\max_a\{R(s,a) + \gamma V(s')\}$, selecting a single optimal action for each state. In contrast, IB-MDP employs an ensemble of Monte Carlo Tree Search runs with different random seeds to explore the policy space more thoroughly. This ensemble approach generates multiple independent policies, aggregates their recommendations, and ranks feature sets based on their selection frequency across the ensemble.

The superior coverage of IB-MDP can be attributed to this fundamental difference in policy generation. While VI methods converge to a single deterministic policy, potentially missing equally valuable alternatives, IB-MDP's ensemble approach allows it to identify multiple high-value feature combinations through independent policy searches. This is particularly valuable in experimental design, where multiple feature combinations might yield similar information gains. The results suggest

that ensemble-based policy generation provides more robust recommendations by identifying these equivalent policies, while deterministic approaches may arbitrarily select among equally valuable options.

H CONCEPTUAL FRAMING OF IB-MDP AS A POMDP

This paragraph outlines how the Implicit Bayesian Markov Decision Process (IB-MDP) framework can be interpreted within the conceptual structure of a Partially Observable Markov Decision Process (POMDP). While standard POMDPs rely on explicitly maintaining and updating a belief state distribution over hidden states, IB-MDP employs a data-driven, similarity-weighted sampling approach. This mechanism serves as a practical heuristic to approximate the belief dynamics inherent in POMDPs, particularly in information-gathering scenarios. Below, we detail the mapping between IB-MDP and POMDP components and elaborate on their conceptual relationship.

H.1 POMDP FRAMEWORK OVERVIEW

A POMDP is formally defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \Omega, P, O, R, \gamma \rangle$, where:

- $\mathcal{S}$: A set of hidden states, unobservable by the agent.
- $\mathcal{A}$: A set of actions available to the agent.
- Ω: A set of possible observations.
- $P(s'|s,a)$: The state transition probability function, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$.
- $O(\omega|s',a)$: The observation probability function, $O : \Omega \times \mathcal{S} \times \mathcal{A} \to [0,1]$.
- $R(s,a)$: The reward function, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$.
- γ: A discount factor ($0 \leq \gamma < 1$).

Since the agent cannot observe $s \in \mathcal{S}$ directly, it maintains a **belief state** $b(s)$, which is a probability distribution over $\mathcal{S}$, representing the agent's belief about the current hidden state. This belief is updated after taking action a and receiving observation ω using Bayes' theorem:

$$b'(s') = \eta \cdot O(\omega|s',a) \sum_{s \in \mathcal{S}} P(s'|s,a) b(s),$$

where η is a normalizing constant ensuring $\sum_{s'} b'(s') = 1$. The term $\sum_{s \in \mathcal{S}} P(s'|s,a)b(s)$ represents the predicted belief before observing ω.

H.2 IB-MDP FRAMEWORK REVISIT

IB-MDP is tailored for scenarios, like experimental design in drug discovery, where the true state (e.g., a compound's complete profile) is only partially known. Instead of an explicit belief state, IB-MDP simulates transitions using historical data:

$$P(s'_{\text{observed}}|s_{\text{observed}},a) = \sum_{i=1}^{N} \frac{w_i(s_{\text{observed}})}{\sum_{j=1}^{N} w_j(s_{\text{observed}})} \cdot \mathbb{I}\left[s'_{\text{observed}} = s_{\text{observed}} \oplus \Delta s(a, D_i)\right].$$

Here:

- s_{observed}: The currently observed state (e.g., measured features).
- $\mathcal{D} = \{D_i\}_{i=1}^{N}$: A historical dataset containing past trajectories or state examples.
- $w_i(s_{\text{observed}}) = \exp\left(-\lambda_w \cdot d(s_{\text{observed}}, D_i)\right)$: Similarity weight between s_{observed} and historical data point D_i.
- $d(\cdot,\cdot)$: A distance metric defined over the observed features.
- $\Delta s(a, D_i)$: The state change (often revealing new feature values) associated with action a in the context of D_i.

- $\oplus$: An operator that updates the observed state, typically by augmenting it with newly revealed information from Δs.
- $\mathbb{I}[\cdot]$: The indicator function.

This mechanism implicitly handles uncertainty by sampling transitions based on similarity to past experiences, avoiding explicit belief representation.

H.3 Mapping IB-MDP to the POMDP Framework

The components of IB-MDP can be mapped conceptually to the POMDP framework:

- **Hidden States ($\mathcal{S}$)**: Corresponds to the true, complete, underlying state of the system, denoted s_{true}. In drug discovery, this is the full set of intrinsic properties of the compound, mostly unknown initially.
- **Actions ($\mathcal{A}$)**: These are directly equivalent – the choices available to the agent (e.g., performing an assay).
- **Observations (Ω)**: In IB-MDP, an explicit observation set Ω isn't defined. Instead, taking action a and simulating the transition via sampling D_i yields a state change $\Delta s(a, D_i)$. The *new information* revealed within Δs (e.g., the value of a newly measured feature) serves the role of the observation ω.
- **Transition Function (P)**: In many IB-MDP applications involving information gathering, the underlying true state s_{true} is static ($P(s'_{\text{true}}|s_{\text{true}}, a) = \mathbb{I}[s'_{\text{true}} = s_{\text{true}}]$). Actions only serve to reveal information. The IB-MDP transition function $P(s'_{\text{observed}}|s_{\text{observed}}, a)$ models the change in the *observed* state based on sampling, which reflects gaining information about the static s_{true}.
- **Belief State (b)**: The POMDP belief $b(s_{\text{true}})$ is approximated in IB-MDP by an **implicit belief representation** consisting of the pair $(s_{\text{observed}}, W(s_{\text{observed}}))$:
  - s_{observed}: The agent's current knowledge manifested as observed features.
  - $W(s_{\text{observed}}) = \{w_i(s_{\text{observed}})\}_{i=1}^{N}$: The vector of similarity weights. This vector defines a distribution over the historical dataset $\mathcal{D}$, implicitly encoding belief about s_{true} by highlighting which past examples are currently considered most relevant or likely.
- **Belief Update**: The POMDP belief b is updated via Bayes' rule. In IB-MDP, the update occurs when action a yields new information, augmenting s_{observed} to s'_{observed}. The agent then **recalculates the similarity weights** $W(s'_{\text{observed}})$. This recalculation acts as a **heuristic belief update**, adjusting the distribution over $\mathcal{D}$ to reflect the new information (implicit observation ω), thereby refining the implicit belief about s_{true}.

H.4 Conceptual Parallel: IB-MDP Update as an Approximate Belief Update

We can illustrate the conceptual parallel between the formal POMDP belief update and the IB-MDP weight recalculation.

H.4.1 POMDP Belief Update (Information Gathering)

For a static true state ($s' = s$), the update simplifies:

$$b'(s_{\text{true}}) = \eta \cdot O(\omega|s_{\text{true}}, a) \cdot b(s_{\text{true}}),$$

The posterior belief is proportional to the prior belief multiplied by the likelihood of observing ω given the state s_{true}.

H.4.2 IB-MDP Implicit Update Steps

The process within an MCTS simulation step or real-world step involves:

1. **Implicit Prior**: The current weights $W(s_{\text{observed}})$ define a distribution over $\mathcal{D}$, acting as a heuristic prior belief about which historical states resemble the true state.

2. **Implicit Observation Likelihood**: An action a is taken. A historical point D_i is sampled based on $W(s_{\text{observed}})$. The resulting state change $\Delta s(a, D_i)$ reveals new information (implicit observation ω). The process inherently favors sampling D_i consistent with likely observations.
3. **Implicit Posterior**: The observed state is updated to $s'_{\text{observed}} = s_{\text{observed}} \oplus \Delta s(a, D_i)$. The weights are recalculated as $W(s'_{\text{observed}})$. This new weight vector represents the updated implicit belief, giving higher weight to historical points that match the augmented s'_{observed} (i.e., are consistent with the observation ω).

While the POMDP update uses the mathematically grounded Bayes' rule, IB-MDP employs a computationally tractable heuristic: recalculating similarity weights based on the augmented observed state. This adjusts the focus on relevant historical data, effectively updating the agent's implicit belief.

H.5 Implications and Conclusion

Framing IB-MDP within the POMDP context yields several insights:

- **Justification for Dynamics**: The changing transition probabilities observed *within* MCTS simulations in IB-MDP are not arbitrary non-stationarity. They reflect the simulation of belief updating inherent in solving POMDPs – as information is gathered (state is augmented), the model used for subsequent predictions naturally changes.
- **Suitability of MCTS**: MCTS is a viable algorithm because it effectively explores the consequences of actions on the (implicit) belief state and associated future rewards. Its ability to handle large state spaces makes it suitable for navigating complex belief representations, even approximate ones.
- **Heuristic Nature**: IB-MDP provides a practical, data-driven approximation to solving a POMDP. Its effectiveness relies on the quality of the historical data $\mathcal{D}$ and the appropriateness of the similarity metric $d(\cdot, \cdot)$ in capturing relevant state information. Formal convergence guarantees depend on the properties of this approximation.
- **Computational Efficiency**: By avoiding explicit belief state representation and Bayesian updates, IB-MDP offers a potentially more scalable approach for complex problems where POMDPs become intractable.

In conclusion, interpreting IB-MDP as an approximate POMDP framework provides strong conceptual grounding. It clarifies the role of state augmentation and weight recalculation as a form of implicit belief updating, justifying the methodology and the application of MCTS for planning under uncertainty.

I Broader Impacts

The IB-MDP framework provides a versatile approach to adaptive decision making with potential benefits beyond preclinical assay scheduling. By integrating historical data, dynamic heuristic belief updates, and ensemble methods, the framework can potentially enhance decision-making efficiency in various fields requiring sequential planning under uncertainty with resource constraints, such as healthcare logistics or adaptive clinical trial design (Coronato et al., 2020). Its ability to handle uncertainty and constraints via data-driven simulation and robust planning makes it a potentially valuable tool for improving outcomes where explicit models are unavailable but historical data exists.

J Limitations

Increasing the ensemble size N_e enhances the accuracy and robustness of the MLASP but increases computational costs, with diminishing returns typical of ensemble methods (Zhou, 2012). The optimal N_e depends on problem complexity and resources. Similarly, the number of MCTS iterations n_{itr} affects planning quality versus compute time. While the ensemble approach generally stabilizes, convergence assessment in complex state spaces remains empirical.

Another important limitation lies in the framework's reliance on the quality and coverage of the historical data $\mathcal{D}$ and the chosen similarity metric. The heuristic belief update's effectiveness depends on these components. Although leveraging historical data helps to integrate similarity-based metrics for decision-making, it may not sufficiently account for novel scenarios in real-world experiments. Future extensions of the IB-MDP framework could incorporate more flexible strategies, such as adaptive similarity metrics (cf. Bellet et al., 2013) or hybrid approaches integrating deep learning components (e.g., for feature extraction or value estimation, cf. Blau et al., 2022), to extrapolate to states not represented in the existing dataset.

Furthermore, the current study lacks direct experimental comparisons with state-of-the-art BRL (e.g., PSRL variants (Osband et al., 2013; Agrawal & Jia, 2017)) or recent BED/RL methods specifically tailored for implicit models or sequential design (Ivanova et al., 2021; Blau et al., 2022). Such benchmarking is essential future work to rigorously assess relative performance.

Finally, while likelihood thresholds (τ) guide the MLASP, further exploration of the interplay between uncertainty reduction ($\mathcal{H}$) and likelihood ($\mathcal{L}$) dynamics could yield insights for refining policy generation and ensemble aggregation, potentially improving performance in complex dynamic environments.