Navigating the Exploitation-Exploration Tradeoff: An Empirical Study of Resource Allocation in Research Labs

Ran Zhuo
Technology and Operations Department, Ross School of Business, University of Michigan, ranzhuo@umich.edu

Balancing exploitation and exploration in resource allocation under uncertainty is a classic theoretical problem. Yet little research has empirically studied how organizations navigate the exploitation-exploration tradeoff in complex real-world situations. To address this gap, this paper introduces a novel setting of structural biology labs, featuring high-frequency, publicly available data on nearly one million discrete experimental trials allocated across 300,000+ research projects from 2000-2015. We model this setting as a stochastic bandit and develop a dynamic structural estimation approach to infer the allocation decision policies that best characterize lab behavior. We find the labs' decision models strongly resemble a simple Upper Confidence Bound (UCB) algorithm, which achieves superior in-sample fit (51–84% of the log-likelihood of the next-best model among the ones we tested with minimal additional parameters) and strong out-of-sample predictive accuracy (73–87% allocation probability for actually allocated trials versus 0.1–0.8% for unallocated ones). Through counterfactual simulations, we demonstrate how to leverage our policy inference results to incrementally evaluate and improve allocation decision making. For example, switching to a readily implementable alternative algorithm could have increased cumulative rewards by up to 28%, while earlier adoption of structured decision-making during these labs' initial pilot phases could have yielded further performance gains, though results vary significantly across labs due to organizational heterogeneity.

Key words: explore-exploit tradeoff, stochastic bandit, resource allocation, policy inference, innovation

1. Introduction

Organizations frequently encounter a fundamental dilemma between exploring new opportunities and exploiting existing knowledge. This tradeoff, formalized by seminal works such as March (1991), represents a central challenge in organizational learning and resource allocation. While theoretical foundations of the exploration–exploitation dilemma—often modeled as a bandit problem—have been extensively studied since Thompson (1933) and Gittins (1979), empirical evidence on how decision-makers actually navigate this tradeoff in complex real-world environments remains limited. This is largely due to challenges in obtaining sufficiently granular data and computational difficulty of modeling dynamic decision processes with large action spaces.

This paper addresses these empirical and methodological gaps. We introduce a novel empirical setting with rich, high-frequency, publicly available data on resource allocation decisions within organizations, and develop a dynamic structural estimation approach to infer allocation decision policies employed by decision-makers. Why is policy inference valuable when the literature has largely focused on designing optimal allocation algorithms (see Lattimore and Szepesvári (2020) for a comprehensive overview)? In practice, organizational decision-making is often shaped by institutional constraints, behavioral frictions, and political considerations that limit the feasibility of wholesale policy changes. Moreover, real-world allocation problems frequently involve complexities that make deriving theoretically optimal policies either highly challenging or outright infeasible. Policy inference enables us to benchmark current behavior and evaluate the marginal benefits of small, targeted policy changes without relying on an optimal benchmark, thereby providing evidence-based recommendations that are more likely to be adopted in practice.

Our empirical setting involves large, publicly funded structural biology labs participating in the Protein Structure Initiative (PSI), a $1.3 billion grant program managed by the National Institutes of Health (NIH) from 2000 to 2015. The core task of these labs—allocating experimental trials to protein structure determination projects—elegantly maps onto a stochastic bandit framework. Each protein molecule represents a distinct project (arm), labs allocate resources through discrete experimental trials (arm pulls), and outcomes are highly uncertain (98% failure rate). PSI labs conducted nearly one million experimental trials across more than 300,000 protein structure determination projects, with allocation decisions observed daily. This provides an ideal empirical context with rich data for evaluating organizational decision-making in a bandit framework.

We develop a likelihood-based policy inference framework within a stochastic bandit setting to identify policy classes that best explain observed allocation decisions. Unlike most bandit literature, which aims to design optimal algorithms, our goal is to infer decision-making policies from behavior without assuming optimality. This approach, commonly referred to as structural estimation, assumes an overarching and invariant structure for a decision problem, then estimates parameters within that structure using data. Its key strength lies in enabling counterfactual analysis,

though its validity depends on the correctness of the assumed structure. In our case, the structure is a stochastic bandit—a natural fit for creative, sequential decision-making processes involving exploration-exploitation trade-offs. We also confirmed with lab managers that their decision-making processes broadly conform to the bandit structure.

We enhance model realism by incorporating institutional knowledge, since standard bandit structures remain relatively simple compared to real-world decision environments. For example, rather than assuming independent projects, we allow project outcomes to depend on a high-dimensional set of molecular properties, generating correlated rewards and learning spillovers among similar projects. While these enhancements increase empirical relevance, they also render the model analytically intractable and make estimation more challenging. Classic dynamic structural methods (e.g., Pakes 1986, Rust 1987, Hotz and Miller 1993) are infeasible here due to the curse of dimensionality.

Our likelihood-based framework addresses the estimation challenge by decomposing the inference problem into three components: (1) recovering the reward function that captures organizational preferences (e.g., research quantity versus impact), (2) identifying the belief-updating model that characterizes how decision-makers learn about reward distributions from past outcomes, and (3) inferring the decision policy that maps beliefs and preferences to allocation decisions. Leveraging detailed institutional knowledge, we estimate belief-updating processes offline, enabling efficient estimation of reward functions and allocation policies from observed behavior. Our approach shares high-level similarities with concurrent work by Ano and Martinez-de Albeniz (2023), though our empirical setting and inference procedure differ.

Empirically, we find that resource allocation in these labs is best described by exploration-oriented models. Specifically, a variant of the UCB algorithm—UCB1 with time-discounted exploration bonuses—consistently outperforms other specifications, including Greedy, Gittins Index, Thompson Sampling, and Explore-Then-Commit, across all labs. This model achieves superior in-sample fit (51–84% of the log-likelihood of the next-best model with minimal additional parameters) and strong out-of-sample predictive accuracy (73–87% allocation probability for actually

allocated trials versus 0.1–0.8% for unallocated ones). The estimated parameters align with institutional context: labs placed positive weight on exploration, favored more recent projects, and responded to NIH directives emphasizing biomedical importance.

Through counterfactual simulations, we evaluate how alternative policies might have performed. For one major lab, adopting an Explore-Then-Commit algorithm could have improved cumulative rewards by up to 28%. In other cases, the inferred policies already performed the best among the readily adoptable policies we tested. Across all labs, Greedy policies consistently underperformed, reaffirming the value of structured exploration. We further show that earlier adoption of structured decision-making during the PSI's initial pilot phase could have yielded additional performance gains. Finally, we find that if labs possess perfect information about each project's reward distribution, cumulative rewards could nearly double across labs—highlighting the value of high-quality historical data.

We conclude by discussing limitations and opportunities for future work. Among these, we do not model the upstream project nomination process that shapes the set of alternatives (or "arms") available to decision-makers—an important but distinct component of organizational resource allocation. Moreover, we cannot fully explain why certain algorithms outperform others in specific labs, given the limited number of labs observed. Nonetheless, we see this study as a foundation for a broader empirical agenda on how organizations learn and allocate resources under uncertainty.

The remainder of the paper is organized as follows. Section 2 develops the theoretical framework for policy inference in stochastic bandit settings. Section 3 introduces the empirical setting of structural biology labs, and Section 4 maps their resource allocation problem to the bandit framework. Section 5 details the likelihood-based inference procedure. Section 6 presents the policy inference results, showing that lab decisions are best explained by an exploration-based model with a time-discounted exploration bonus. Section 7 uses the inferred policy to counterfactually evaluate the impact of incremental improvements in allocation. Section 8 concludes with limitations and future directions.

2. Policy Inference Framework for Stochastic Bandits

2.1. Bandit Problem Setup

Drawing on the canonical stochastic bandit framework in Lattimore and Szepesvári (2020), we develop a theoretical framework for policy inference that underpins our empirical analysis of exploitation-exploration decisions in organizational resource allocation.

In this framework, resource allocation is modeled as a sequential game between the decision-maker and the stochastic environment over a horizon of T periods. In each period $t \in \{1, 2, ..., T\}$, the environment first reveals a context $c_t \in C$, representing observable features relevant to the decision at time t. The decision-maker then selects an action a_t (or "pulls an arm") from a finite set of available actions $\mathcal{A}_t$. Crucially, the decision-maker cannot foresee future outcomes and must select a_t based solely on the observed context c_t and the history up to period $t-1$, denoted $H_{t-1} = (c_1, a_1, x_1, \ldots, c_{t-1}, a_{t-1}, x_{t-1})$. A policy π governs this decision process by mapping the current context and history to a probability distribution over actions, such that under policy π, the conditional distribution of action a_t given H_{t-1} and c_t is $\pi_t(\cdot \mid H_{t-1}, c_t)$. The environment then generates a reward $x_t \in \mathbb{R}$ sampled from the conditional probability distribution $P_{a_t}(\cdot \mid H_{t-1}, c_t)$ and reveals x_t to the decision-maker.

The decision-maker's objective is to maximize the cumulative reward $\sum_{t=1}^{T} x_t$ across the time horizon. Achieving this requires adopting a policy that effectively learns from past outcomes and balances the trade-off between exploiting known high-reward arms and exploring uncertain alternatives. The environment generates rewards according to the history, context, and action, providing feedback that enables the policy to adapt and improve decision-making over time.

This stochastic contextual bandit formulation nests the standard stochastic multi-armed bandit as a special case when the context c_t is fixed or uninformative, and maps onto many real-world organizational decision problems. In product development contexts analyzed by Ano and Martinez-de Albeniz (2023), for example, managers balance launching products in well-understood categories (exploitation) versus testing novel categories with uncertain potential (exploration). Similarly, in our empirical setting, labs allocate experimental trials across molecular targets, facing analogous trade-offs.

2.2. Research Focus: Policy Inference

Unlike the traditional bandit literature, which focuses on designing optimal algorithms for various reward distributions and arm-pulling scenarios, our research centers on *policy inference*—recovering the underlying decision policy from observed actions and outcomes. This requires identifying how decision-makers use contextual information and historical data to learn about the environment and reward distributions, as well as determining both the policy class (e.g., UCB, Thompson Sampling) and its parameters (e.g., exploration rates) that are employed.

Our approach draws on the extensive structural estimation literature (see Hortaçsu and Joo (2023) for an overview), which posits an overarching, invariant structure for a decision problem. Within such a structure, one can hypothesize alternative behavioral models, estimate their parameters from data, and assess their fit and explanatory power. A major strength of this approach is its capacity for counterfactual analysis (e.g., evaluating how outcomes would change under different behavioral models) though its validity hinges on the correctness of the assumed structure.

In our setting, the assumed structure is a stochastic bandit, a natural choice for creative, sequential decision-making processes involving exploration–exploitation trade-offs. Discussions with lab managers confirmed that their decision processes align with this structure. Within it, we specify and test a range of behavioral models for allocation, from established bandit algorithms to reduced-form specifications that capture decision patterns in a theory-free manner.

The central assumption enabling policy inference here is *revealed preference*: decision-makers reveal what they value through their choices—a foundational concept in decision theory (see Varian et al. (2006) for an overview). At a high level, we infer the extent to which labs valued exploration by observing how often they allocated trials to projects with no clear immediate benefit—for example, those with uncertain success probabilities or unrelated to stated NIH priorities. This inferred value reflects revealed preferences, which need not correspond to optimal behavior. To our knowledge, the only other paper to study policy inference in a bandit setting is Ano and Martinez-de Albeniz (2023), which shares methodological similarities with our approach but differs in the model assumptions enabling parameter identification.

Policy inference is related to—but distinct from—inverse reinforcement learning (IRL). IRL typically assumes optimal behavior in a given setting and aims to recover the reward function that would make the observed actions optimal (Ng et al. 2000). In contrast, policy inference makes no assumption of optimality (Chan et al. 2022). Instead, it seeks the policy most likely to have generated the observed action-reward sequences (similar to Hüyük et al. (2021)), whether or not that policy is optimal, near-optimal, or flawed. This distinction is crucial in organizational settings, where decisions often rely on intuition, heuristics, or qualitative rules rather than formal optimization. Moreover, real-world resource allocation problems are usually too complex for analytically optimal solutions. By inferring and analyzing the actual policies implemented, we can reveal behavioral and institutional influences shaping decisions and identify opportunities to improve organizational processes—either by proposing alternative policies or redesigning the decision environment.

To avoid confusion, we distinguish *policy inference* from several related concepts in the bandit literature. *Inference* typically refers to estimating the true mean rewards of arms from collected data. This is challenging in bandit settings because arms are adaptively selected, producing non-i.i.d. reward data. As a result, sample averaging—common in bandit algorithms—can be biased and statistically unreliable (Dimakopoulou et al. 2021, Kalvit and Zeevi 2021, Simchi-Levi and Wang 2023). *Policy evaluation* builds on accurate *inference* and involves estimating the expected cumulative (or average) reward of a known policy. *Off-policy evaluation*, in particular, aims to evaluate a counterfactual policy using data generated by a different policy. *Policy learning*, which relies on reliable evaluation, focuses on optimizing policies to improve decision-making over time (Kallus et al. 2022). Later in the paper, we demonstrate how our policy inference results support simulation-based off-policy evaluation and improvement.

2.3. Decomposing the Empirical Challenges for Policy Inference

Inferring the decision policy in a bandit setting involves disentangling three fundamental components:

1. **Reward function form, x_t:** This captures the decision-maker's preferences, which may involve multiple objectives. For example, a revenue-focused firm might set x_t equal to period

revenue, whereas a product development firm could optimize a weighted combination of revenue and market share. Similarly, research labs often balance scientific impact, publication volume, and funding requirements, leading to a more complex reward function. Additionally, x_t may depend on the context c_t, reflecting how outcomes vary with external conditions or task features.

2. **Learning model, $\widehat{P}_t$:** This represents how the decision-maker's beliefs about the reward distribution evolve as they observe new rewards obtained from arm pulls. It is important to distinguish $\widehat{P}_t$ from P_t, the true (but unknown) reward-generating process for each arm determined by the environment; policy inference requires *only* understanding how decision-makers perceive and update their beliefs about rewards, rather than the actual reward dynamics.

3. **Decision policy form, π:** This defines how the decision-maker translates beliefs and preferences into action choices. It includes the policy class (e.g., UCB, Thompson Sampling) and parameters (such as exploration rates). Crucially, the policy is not assumed to be optimal.

To illustrate how these three components form a decision policy, consider an agent employing the widely used UCB1 algorithm (Auer et al. 2002) for arm selection. The agent first needs to specify the reward function for each arm. For example, imagine a firm deciding which product category (arm) to select for launching a new product (pull), with interest in both revenue and market share. The reward function for each potential product could be expressed as $x_i = 0.5 \times \text{revenue}_i + 0.5 \times \text{market share}_i$. The firm then forms beliefs about the reward distribution based on past outcomes. Specifically, the UCB1 algorithm requires estimating the expected reward using the empirical mean reward $\hat{x}_i(t-1)$ from previously launched products in that category. The UCB1 algorithm then maps these preferences and beliefs into a decision rule by computing a confidence-adjusted index value for each arm i at each decision point t:

$$V_{it} = \begin{cases} \infty & \text{if } J_i(t-1) = 0, \\ \hat{x}_i(t-1) + \sqrt{\frac{2\ln(N_t)}{J_i(t-1)}} & \text{otherwise,} \end{cases} \tag{1}$$

where $J_i(t-1)$ is the number of times arm i has been pulled up to period $t-1$, and N_t is the total number of pulls across all arms up to that point. The algorithm then selects the arm with the highest index value in each period.

Inferring decision policy components in a bandit setting requires highly granular sequential data. At minimum, we need:

- The contexts relevant to the action choice in each period, $(c_1, c_2, ..., c_T)$
- The actual actions taken in each period $(a_1, a_2, ..., a_T)$
- The set of available actions to choose from in each period $(\mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_T)$
- The realized outcomes contributing to rewards $(x_1, x_2, \ldots, x_T)$, potentially reflecting multiple objectives

However, sequential choice data alone does not permit separate identification of the three key components without additional institutional knowledge. To see this, consider again a product development firm selecting new products: if we observe seemingly random revenue patterns across products even in later periods, multiple interpretations are possible. The decision-maker might: (1) have objectives beyond revenue maximization (affecting the reward function); (2) employ an ineffective learning model that fails to predict promising projects based on past outcomes (affecting the learning mechanism); or (3) mistakenly maintain high exploration rates (affecting policy parameters). In such cases, the reward function, learning mechanism, and policy parameters are not separately identified from observational data alone. If, however, institutional knowledge confirms that the firm prioritizes revenue maximization exclusively, and choice patterns show resources being allocated toward well-tested product categories, we can infer that their learning model is suboptimal. This illustrates how domain-specific knowledge provides crucial identification constraints for policy inference.

In this paper, we leverage institutional knowledge about decision-makers' belief updating processes, enabling us to separately model their learning mechanism with reasonable fidelity. While we cannot directly observe agents' precise trade-offs between competing objectives, we know which variables they considered important. This allows us to specify the reward function as a linear combination of these variables with unknown weights. We therefore estimate the learning model offline, then use the estimated learning model and choice data to jointly estimate both the reward

function weights and policy parameters. This approach differs from Ano and Martinez-de Albeniz (2023), who assumed the firm has a singular revenue maximization objective but lacked institutional detail on belief updating. They instead jointly estimated the learning model and decision policy parameters using choice data.

A remaining question is how to evaluate whether our inferred policy accurately captures the actual decision-making process. We employ two complementary assessment criteria: (1) in-sample fit, measured by the likelihood of observed actions under the inferred policy parameters, and (2) out-of-sample predictive performance, where we estimate policies using only an initial period of observed history and then evaluate the model's ability to predict allocation decisions beyond the estimation period. Together, these measures provide evidence of both the inferred policy's explanatory power and generalizability beyond the estimation sample.

3. Empirical Setting

Our empirical analysis examines decision-making processes within major structural biology labs funded through the Protein Structure Initiative (PSI)—a large-scale scientific program that operated from 2000 to 2015 with $1.3 billion in NIH funding.

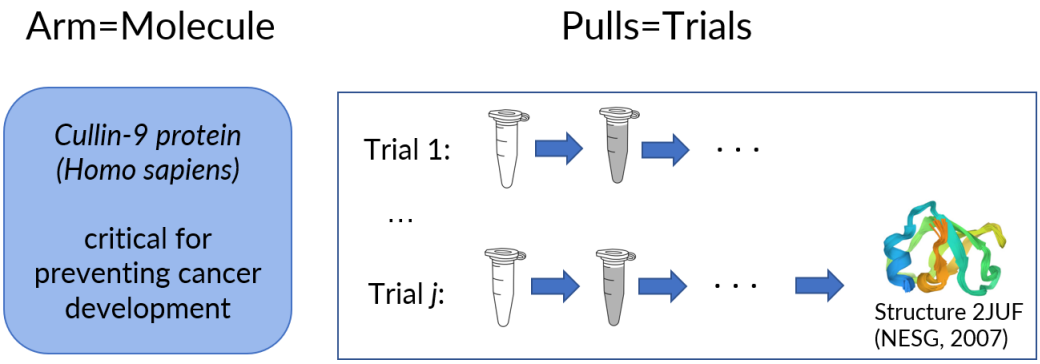
3.1. Scientific Background: Structural Biology

Structural biology is the field devoted to determining the three-dimensional structures of protein molecules (see Figure 1 for an example). Proteins are long amino acid chains that fold into shapes enabling specific functions—much like a key fitting into a lock. These structures serve as blueprints for drug design, driving innovations from targeted cancer therapeutics (Van Montfort and Workman 2017) to COVID-19 vaccines (Wrapp et al. 2020), and have earned the field over a dozen Nobel Prizes (Hill and Stein 2025).

The work within a structural biology lab constitutes a dynamic sequential decision process that closely resembles a classical bandit problem. At each point in time, the lab has an array of proteins to choose from, with each candidate protein representing an "arm" that can be pulled. The lab maintains beliefs about each protein's probability of success and potential reward if successful. Resources are allocated to projects in discrete, countable units—individual experimental

trials—which map naturally to arm pulls in a bandit. The lab chooses which arms to pull, observes the results, updates its beliefs, and repeats the process.

Figure 1 **Protein molecule (arm) and experimental trial (pull) in structural biology research**



Note: Structure determined by Northeast Structural Genomics Consortium (NESG) (Kaustov et al. 2007).

Each experimental trial represents an independent Bernoulli process with binary outcomes: success or failure. Conducting a trial involves multiple sequential steps, much like baking a cake, where failure at any stage requires starting the entire process over. However, unlike baking—where mastering a particular step typically ensures consistent success at that step in future attempts—structural biology trials are highly unpredictable endeavors that do not reliably build upon previous progress. Even if one trial nearly succeeds but fails in the final procedure, the subsequent trial on the identical protein might fail at the very first step. As noted by Chruszcz et al. (2008), "the success of any or all individual steps does not guarantee the success of the overall process... requires a significant amount of work and much luck." This inherent uncertainty is reflected in the dataset examined, where 98% of trials failed to produce any structure.

Multiple trials on the same project can be conducted either in parallel or sequentially. The first successful trial for a project yields a reward—a multi-dimensional benefit to the lab and society, including relevance to disease therapies, understanding of human biology, and publication and citation potential. Failed trials produce no immediate reward, and subsequent successes after the first yield no additional reward since duplicate structures contribute little new scientific value.

The true probability of success (p) for any trial can be influenced by several factors: previous trials with the same molecule, previous trials on similar molecules sharing key molecular properties,

and inherent characteristics of the molecule itself (for example, shorter molecules are typically easier to analyze than longer ones). Previous trials can increase the true success probability (p) if scientists gained technical expertise from earlier attempts. Previous trial outcomes from the same or similar molecules may also help labs learn and refine their beliefs ($\widehat{p}$) about success probabilities, bringing $\widehat{p}$ closer to p and narrowing its confidence intervals.

This creates the classic exploration-exploitation tradeoff that defines bandit problems: allocating trials to less-tested projects helps researchers learn about their success probabilities (and those of similar projects sharing key characteristics), but diverts resources from projects already known to have high success rates.[1]

3.2. Institutional Background: Protein Structure Initiative (PSI)

The PSI was a major NIH program operating from 2000 to 2015, comprising four large labs and numerous smaller ones with substantial variation in scale. Unlike investigator-initiated R01 grants awarded competitively to individual labs, the PSI operated under U01 cooperative agreements. These NIH-initiated agreements promoted structured collaboration and sustained interaction between the agency and researchers, fostering information sharing and expanding research into molecules that traditional R01-funded labs might have lacked incentives to pursue. PSI labs were required to collect detailed data on trial allocation and outcomes across projects, all made publicly available in real time through the TargetTrack database (Berman et al. 2017) as trials were allocated and performed.

The PSI evolved through three phases reflecting shifts in NIH priorities. During the Pilot Phase (2000–2004), the NIH avoided setting production targets, encouraging labs to explore and develop scalable tools and processes (NIGMS 2008). With hundreds of millions of known protein molecules, the initial narrowing of projects (i.e., choosing "arms" available for pull) was as important as

[1] A related but distinct question is whether the labs should have the option to terminate a trial early or adapt its effort midstream. We assume a trial ended if and only if it failed—i.e., labs did not actively intervene mid-trial. This is supported by data: 15% of trials list termination reasons, mostly exogenous (e.g., "expression failed," "poor diffraction"). Strategic terminations (e.g., "duplicate target found") were rare, and many such trials actually continued and succeeded. While mid-trial intervention may be theoretically optimal, modeling such behavior would require embedding an optimal stopping problem within a dynamic allocation problem, a complexity we leave to future research.

deciding which arms to pull. Unfortunately, reasons for initial project inclusion and trial allocations were often noted as "ad hoc" in TargetTrack during this phase, reflecting inconsistent focus and unstructured decision-making.

The Production Phase (2005–2008) began when the NIH established annual production targets of 200 structures per large lab (NIGMS 2004). From this phase onward, the NIH implemented regular performance evaluations[2] based on multiple metrics including structure quantity, novelty, biomedical relevance, and specific protein categories (human, eukaryotic[3], and membrane proteins[4]). This oversight helped mitigate principal-agent problems by aligning lab incentives with stated agency priorities.

The initial narrowing of projects was also refined during this phase through a separate, upstream process managed by the NIH[5] via three channels: (1) a centralized planning committee periodically assigned molecules based on bioinformatics; (2) the biomedical research community could nominate molecules of interest, which the NIH then assigned to labs; and (3) individual labs could propose their own targets, subject to NIH approval (Berman et al. 2017). Once a project entered a lab's choice set, it remained until it succeeded. Occasionally, projects trialed in the early years of the PSI were revisited much later, indicating no formal abandonment.

Large labs began describing their trial allocation approach as "high-throughput" during this phase: initially assigning one trial to many projects, then allocating additional trials to those deemed important or promising based on earlier outcomes. Machine learning methods for predicting project success probabilities also emerged during this period. The shared TargetTrack database enabled labs to quickly process large sets of historical trial data as new trials were performed and outcomes observed, and to use machine learning models trained on historical data to predict

[2]Archived metrics are available at http://targetdb.pdb.org on the Internet Archive.

[3]Eukaryotes are organisms whose cells have a nucleus containing DNA organized into chromosomes. This includes all living organisms except bacteria and archaea.

[4]Membrane proteins are found in the cell membrane and are particularly challenging for structure determination due to their physicochemical properties.

[5]This contrasts with the optimal stopping problem in McCardle et al. (2018), where firms working on the projects decide whether to abandon existing projects and search for new ones.

project success probabilities. These practices are thoroughly documented in multiple publications (Slabinski et al. 2007a,b, Jaroszewski et al. 2008, Price et al. 2009, Babnigg and Joachimiak 2010, Jahandideh et al. 2014).

By mid-2008, concerns over limited attention to biomedically important projects sparked debate within the scientific community (Petsko 2007). In response, the Biomedical Phase (2009–2015) shifted NIH priorities toward biomedically important projects and increased collaboration with external researchers to help identify and include them in the labs' choice sets (NIGMS 2009), while maintaining the same production targets. The structured "high-throughput" decision-making process based on NIH evaluation metrics and machine learning continued throughout this phase.

3.3. Data and Summary Statistics: Significant Heterogeneity Across Labs

The primary dataset for this paper comes from the TargetTrack database (Berman et al. 2017), which contains 961,260 unique experimental trials conducted on 335,553 distinct protein molecules by PSI labs between 2000 and 2015. Our analysis focuses on the four largest labs—Joint Center for Structural Genomics (JCSG) in California, Midwest Center for Structural Genomics (MCSG) in Illinois, Northeast Structural Genomics Consortium (NESG) in New Jersey, and New York Structural Genomics Research Consortium (NYSGRC)—which together accounted for 71% of projects and 85% of documented trials. For these labs, allocation decisions were recorded daily, including the start date of each trial and whether it successfully produced a structure. While TargetTrack includes data from many smaller labs, we focus on these four due to their substantially higher data quality and completeness.

Table 1 summarizes project, trial, and outcome characteristics for these labs, revealing substantial heterogeneity despite the nominally uniform NIH processes for initial project narrowing and productivity evaluation. Each lab exhibited distinct patterns: JCSG conducted an especially large number of total trials and trials per project; MCSG received notably higher funding and more projects; NESG recorded the lowest success rates at the project level and the fewest structures determined; and NYSGRC had a few projects with exceptionally high trial counts, many times the

Table 1 Summary Statistics of Major PSI Labs' Trial Allocation and Outcomes, 2000–2015

	JCSG	MCSG	NESG	NYSGRC*
Total funding (in 2015 dollars, millions)	177	218	170	159
Project-level characteristics				
No. of projects (molecules) assigned	40,881	77,200	59,946	59,734
% successful projects	3.7	2.9	1.8	2.2
% projects labeled biomedically important	66.0	25.0	21.3	70.6
% projects labeled novel	93.6	35.1	83.9	80.5
% projects related to human proteins	71.1	44.0	77.9	50.6
% projects related to eukaryotes	59.7	54.2	71.1	56.4
% projects related to membrane proteins	11.8	18.7	25.1	17.1
Trial-level characteristics				
No. of trials allocated	378,363	158,727	133,240	149,207
Avg. no. of trials per day	64.8	27.1	22.8	25.5
% successful trials	0.4	1.8	0.9	3.8
Avg. no. of trials per project	9.3	2.1	2.2	2.5
Std. dev. of trials per project	28.4	8.6	3.2	24.6
Min. no. of trials per project	1.0	1.0	1.0	1.0
25th percentile of trials per project	1.0	1.0	1.0	1.0
50th percentile (median) of trials per project	1.0	1.0	1.0	1.0
75th percentile of trials per project	8.0	1.0	2.0	2.0
Max. no. of trials per project	802.0	1,465.0	93.0	5,410.0
% trials on biomedically important projects	71.0	56.8	31.6	74.6
% trials on novel projects	92.4	28.6	76.0	76.2
% trials on human protein projects	73.8	47.2	84.7	69.7
% trials on eukaryotic projects	55.1	54.2	74.6	65.4
% trials on membrane projects	9.4	19.0	26.6	29.4
Outcome-level characteristics				
No. of unique structures successfully determined	1,509	2,203	1,063	1,334
% structures biomedically important	67.5	54.2	31.9	48.7
% structures novel	95.5	21.4	80.8	66.0
% structures related to human proteins	65.3	43.5	74.4	49.8
% structures related to eukaryotes	51.1	54.6	63.6	56.1
% structures of membrane proteins	8.9	11.4	16.0	14.7

Note: See Appendix A for variable construction details. The number of successful projects may not equal the number of successful trials, as multiple successful trials for the same project can yield duplicate structures. *NYSGRC data should be interpreted cautiously due to chronological inconsistencies. Over half of structure-producing trials have missing or misordered dates (e.g., structure publication preceding trial allocation). To preserve a complete trial history, we attempted to correct these by reordering dates to reflect a plausible sequence of events for each trial; however, the extent of inconsistencies may still limit the reliability of NYSGRC data.

maximum observed elsewhere. Such differences likely reflect operational realities and idiosyncrasies. For instance, the NIH's practice of assigning community-nominated projects to geographically proximate labs to facilitate collaboration could influence the number of projects allocated to each lab. Likewise, differences in equipment configurations—such as varying access to X-ray crystallog-

raphy, NMR, and cryo-EM—could shape each lab's trial capacity.[6] These disparities suggest that each lab faced a distinct resource allocation problem, with differing resource levels, numbers of arms, and regions of the reward distribution to explore. A one-size-fits-all policy recommendation is therefore unlikely to be effective.

Notably, across all labs, every project (or molecule) received at least one trial, consistent with prescriptions from canonical bandit algorithms such as Explore-Then-Commit and UCB1.

In addition to the trial-level outcomes reported in Table 1, TargetTrack records intermediate outcomes (success or failure) for each procedure within a trial, as well as procedure durations. We incorporate these data into our analysis; see Appendices B and D for details. We also compiled hundreds of molecular properties (e.g., molecular weight, water affinity) for PSI projects from publicly available sources, which we use to model the labs' learning and belief-updating processes. These variables are described in Appendix A.

4. Mapping the Empirical Setting Into a Stochastic Bandit Framework

This section elaborates on how structural biology labs' trial allocation problem maps into the stochastic bandit framework from Section 2.1. Labs, rather than individual scientists, are treated as the decision-makers due to limited scientist-level data for some labs. We focus on analyzing the Production Phase (2005–2008) and Biomedical Phase (2009–2015) of the PSI, excluding the Pilot Phase (2000–2004) whose ad hoc decision-making lacked sufficient structure for meaningful analysis. The time horizon T thus spans 2005–2015, discretized daily to match trial records, so $t \in \{2005/01/01, 2005/01/02, ..., 2015/12/31\}$.

Projects represent arms, while trials constitute pulls. Subject to daily capacity constraints n_t, labs can allocate multiple trials across different projects, multiple trials to a single project, or any combination thereof. The set of arms available at time t, denoted as $K_t(H_{t-1})$, depends on history and evolves dynamically as successful projects exit and new projects are identified through the

[6] Since trials typically followed standardized procedures within each lab, it is reasonable to assume relatively consistent per-trial costs within labs. Conversations with NIH program officers confirmed that cost-per-trial was neither recorded nor its variation systematically considered in allocation decisions.

three NIH mechanisms. The capacity n_t is assumed to be exogenous, determined by each lab's available scientists and equipment, and is set equal to the observed number of trials allocated each day. For each arm $i \in K_t(H_{t-1})$, the possible action is $a_{i,t} \in \{0, 1, 2..., n_t\}$ (i.e. allocating up to n_t trials), with total trials respecting the capacity constraint: $\sum_{i \in K_t(H_{t-1})} a_{i,t} = n_t$. The action space (set of valid trial allocations) at time t is $\mathcal{A}_t = \{(a_{1,t}, a_{2,t}, \ldots, a_{|K_t|,t}) \in \mathbb{Z}_{\geq 0}^{|K_t|} : \sum_{i \in K_t(H_{t-1})} a_{i,t} = n_t\}$.

Crucially, labs cannot anticipate future outcomes when allocating trials, so choices at time t (a_t) depend solely on the contextual properties of each molecule (c_t) and the history of past trials and rewards (H_{t-1}). The context c_t contains hundreds of factors that can potentially drive a structural determination project's probability of success and reward, thereby influencing allocation decisions. These factors include time-invariant molecular properties (e.g., molecular weight or water affinity) and time-varying ones (e.g., molecule-specific scientific progress outside the PSI labs that increases its success probability). Some components of c_t may be extracted from the history—for example, experience from previous trials on a molecule (or similar molecules) may alter its underlying success probability p, while past trial outcomes may refine labs' beliefs $\widehat{p}$ about that probability. After labs allocate their daily trials, nature generates rewards from the conditional distribution $P_{a_t}(\cdot \mid H_{t-1}, c_t)$. Each trial's reward is individually observable and reflects multiple NIH evaluation criteria, including human relevance and disease association.

Let x_{ijt} denote the reward from the j-th trial of project i at time t. Each lab's objective is to maximize total reward over the T-period horizon by choosing trial allocations:

$$\max_{a_1 \in \mathcal{A}_1, a_2 \in \mathcal{A}_2, \ldots, a_T \in \mathcal{A}_T} \sum_{t=1}^{T} \sum_{i \in K_t(H_{t-1})} \sum_{j=1}^{a_{i,t}} x_{ijt} \tag{2}$$

This empirical setting presents several complexities beyond the classic bandit problem, and the theoretical literature has yet to provide an algorithm that allocates trials optimally under such conditions. First, our setting is a multiple-play semi-bandit where labs allocate multiple trials across different projects daily, subject to capacity constraints, with each trial outcome individually observable. Second, the problem features dynamic arm availability: the set of available projects $K_t(H_{t-1})$ changes as successful projects leave and new ones are introduced, creating expanding

and sleeping arms. Third, general scientific advances may cause time-dependent changes in success probabilities independent of history, context, or actions—characteristic of restless bandit problems. Finally, projects with similar molecular properties exhibit correlated success rates (e.g., shorter molecules are often easier) and learning spillovers—observing one project's outcomes informs beliefs about similar projects—making this a contextual bandit problem.

In the remainder of this section, we specify how we model the three key components of the decision policy: the reward function, the learning model, and the decision policy form (discussed in Section 2.3).

4.1. Parameterization of the Reward Function x_t

We define p_{it} as the probability of success for a trial of project i on day t—a random variable determined by the environment (nature) and unknown to the decision-maker (the lab). However, a trial's probability of success differs from its probability of *payoff*: only the first successful trial for a project yields a reward, while subsequent successes produce duplicate structures with no additional reward. The payoff probability for trial j of project i at time t, given history H_{t-1} and context c_t, is:

$$q_{ijt}(p_{it} \mid H_{t-1}, c_t) = (1 - p_{it})^{m_{i,t}(H_{t-1})+j-1} p_{it}. \tag{3}$$

where $m_{i,t}(H_{t-1})$ counts the number of ongoing trials for project i started before t whose outcomes are not yet observed. A successful trial j yields a reward only if all $m_{i,t}(H_{t-1})$ ongoing trials and all earlier trials $1, 2, ..., j-1$ allocated that same day fail. When $m_{i,t}(H_{t-1}) = 0$, trial $j = 1$'s probability of payoff reduces to p_{it}.

We model the reward for trial j of project i at time t as a Bernoulli random variable:

$$x_{ijt} \sim Bernoulli(q_{ijt}(p_{it} \mid H_{t-1}, c_t)), \tag{4}$$

with

$$x_{ijt}(p_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) = \begin{cases} c_{it} \cdot \boldsymbol{\theta} & \text{with probability } q_{ijt}(p_{it} \mid H_{t-1}, c_t) \\ 0 & \text{with probability } 1 - q_{ijt}(p_{it} \mid H_{t-1}, c_t). \end{cases} \tag{5}$$

$c_{it} \cdot \boldsymbol{\theta}$ represents the reward the lab receives when a trial pays off, and it is a deterministic function of project i's characteristics c_{it}. Given close NIH oversight, we assume that the characteristics c_{it}

determining lab rewards are the same criteria used by the NIH for productivity evaluation and remain fixed within each PSI phase. These multiple criteria are assumed to be combined linearly through a weight vector $\boldsymbol{\theta}$, where each weight reflects the lab's preference for the corresponding characteristic and is known to the lab (though unknown to us). We therefore specify $c_{it} \cdot \boldsymbol{\theta}$ as

$$\begin{aligned} c_{it} \cdot \boldsymbol{\theta} =& 1 \cdot \theta_{quant} + novel_i \cdot \theta_{novel} + prevStructZ_{it} \cdot \theta_{prevStructZ} + biomed_i \cdot \theta_{biomed} \\ &+ prevPubZ_{it} \cdot \theta_{prevPubZ} + human_i \cdot \theta_{human} + eukaryote_i \cdot \theta_{eukaryote} + membrane_i \cdot \theta_{membrane}. \end{aligned} \tag{6}$$

Whenever a trial pays off, the lab is assumed to receive a baseline reward θ_{quant} plus additional amounts depending on the other characteristics of the project. The weight θ_{novel} is assigned to projects declared as novel ($novel_i$). Regarding novelty, the NIH also emphasized the high value of structures from protein families with few or no previously published structures. To capture this priority, we measure how many structures have been published within each molecule's protein family by each given year. The weight $\theta_{prevStructZ}$ is assigned to $prevStructZ_{it}$, where $prevStructZ_{it}$ indicates how many standard deviations the structure count for molecule i's protein family deviates from the mean structure count across all protein families in that year. The weight θ_{biomed} is assigned to projects declared as biomedically important ($biomed_i$). As with previously published structures, it is important to assess how a molecule's biomedical relevance changes over time. To capture this, we measure the number of publications across all biomedical research fields (beyond structures) related to molecule i by each given year. The weight $\theta_{prevPubZ}$ is assigned to $prevPubZ_{it}$, where $prevPubZ_{it}$ indicates how many standard deviations molecule i's publication count deviates from the annual mean. Additionally, θ_{human} is the weight for projects related to humans ($human_i$), $\theta_{eukaryote}$ is the weight for projects related to eukaryotes ($eukaryote_i$), and $\theta_{membrane}$ is the weight for projects on membrane proteins ($membrane_i$).

4.2. Specification for the Learning Model $\widehat{P}_t$

While the expression $x_{ijt}(p_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t)$ specifies the reward function, the decision-maker (the lab) lacks perfect information about all its components: the lab knows its preferences, the context (including relevant molecular properties), and the history, but not the project's true probability of

success p_{it}. Instead, it forms beliefs $\widehat{p}_{it}$ about this probability based on the context and history.[7] The lab therefore makes allocation decisions based on its *beliefs* about the reward:

$$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) = c_{it} \cdot \boldsymbol{\theta} \cdot q_{ijt}(\widehat{p}_{it} \mid H_{t-1}, c_t). \tag{7}$$

rather than on the *true* reward $x_{ijt}(p_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t)$.

Our model for belief formation and updating of $\widehat{p}_{it}$ replicates how major PSI labs used supervised machine learning to predict trial success probabilities from molecular properties and trial history data. We reconstructed the training features used by the labs, comprising hundreds of molecular properties predictive of trial success—from physicochemical characteristics to NIH-assigned labels of novelty and biomedical importance.[8] For the choice of machine learning algorithm, we implemented random forest (RF) models based on Jahandideh et al. (2014),[9] which found that random forests consistently outperformed alternatives in predicting trial success probabilities.[10]

To capture belief updating dynamics as new trials were performed and outcomes observed, we trained a chronological sequence of models: For each day t, model $\widehat{P}_t(\cdot \mid H_{t-1}, c_t)$ was trained using *only* trial history available before t, with trial outcomes (success/failure) as the target variable and molecular properties as features. Since the model is a random forest composed of independently trained decision trees, each tree generates an individual prediction of a project's success probability $\widehat{p}_{it}^{(ntree)}$ at time t based on its molecular properties. These predictions collectively form a posterior distribution representing the lab's belief about the project's success probability, with mean $\widehat{p}_{it}$ and variance $\widehat{Varp}_{it}$, which can be incorporated into Equation (7) to compute the lab's belief about the project's reward. By training exclusively on trial data preceding t, we ensure that our reconstructed machine learning models incorporate only the information available to decision-makers at each

[7] In the absence of any history, the lab's belief about p_{it} in the first period would be based on a prior. This situation does not arise in our setting: at the initial period $t = 2005/01/01$, labs had access to trial history from the Pilot Phase of the PSI, which informed their beliefs.

[8] A comprehensive list of these features appears in Appendix A.4.

[9] Implementation details are provided in Appendix B.

[10] According to the series of papers the labs published, various algorithms were tested over time, including logistic regression, support vector machines, and random forests. A lab project coordinator we interviewed noted that the growing trial history data had a greater impact on prediction accuracy than the specific choice of algorithm.

point in time, preventing any leakage of future outcomes into past predictions. This approach also mirrors the actual belief updating process, as the series of papers documenting these machine learning models progressively expanded the trial history data used for training over time.

The key feature of learning in this setting is cross-molecular information spillovers rather than just within-molecule updating. Each historical trial provides a data point linking molecular properties to success or failure. The machine learning model learns this mapping: $\widehat{P}_t : H_{t-1}, c_t \to \widehat{p}_{it}$, allowing any molecule's predicted success probability $\widehat{p}_{it}$ to be informed not only by its own trial history but also by the histories of molecules with similar properties. Each failure reveals which molecular properties correlate with poor outcomes, while each success increases $\widehat{p}_{it}$ for molecules sharing similar properties. This makes exploring less-trialed molecules particularly valuable for expanding the feature space and improving out-of-sample predictions.

We note that, while we have made every effort to replicate the labs' actual beliefs, our replication may not be perfect.[11] For instance, constructing some features relies on software packages that have since become obsolete, so we have used the closest available substitutes. However, we do not anticipate that any discrepancies will systematically bias the estimates from our policy inference. Errors in replicating $\widehat{p}_{it}$ and $\widehat{Varp}_{it}$ relative to the labs' actual beliefs will bias our estimates if they are systematically correlated with allocation decisions due to omitted variables. To mitigate this risk, we have included all features documented in the labs' machine learning approaches as well as all NIH evaluation metrics in our replication of these models.

4.3. Parameterization of the Decision Policy π

The allocation policy maps the lab's preferences and beliefs about reward distributions into trial allocation decisions. Discussions with NIH program officers and a lab project coordinator indicated that their "high-throughput" approach was heuristic (i.e. not guided by formal algorithms) but informed by historical outcomes. Their decision process appeared consistent with insights from the bandit literature—particularly the need to explore and maintain optimism under uncertainty—though the optimality of their approach remains unclear. Given the heuristic nature of

[11] All sources of potential discrepancies we are aware of are documented in Appendix B.

the decision process and the existence of multiple parameterizations consistent with the "high-throughput" characterization, we specify a range of behavioral models for the allocation policy to assess their fit to each lab's choice data. These specifications include both theoretically motivated parameterizations from classic bandit algorithms[12] and empirically motivated reduced-form specifications that serve as flexible benchmarks. We focus on simpler models grounded in basic statistical reasoning rather than on complex but theoretically appealing policies (e.g., Kallus et al. (2022), Si et al. (2023)), as we do not expect the labs' decision policies to reflect high statistical or computational sophistication.

All models we consider are *index policies*—approaches that compute an "index" value for each option and allocate to those with the highest indices. This framework underlies many well-known bandit algorithms.[13] For an index policy to be optimal, the bandit problem must satisfy *indexability*—the condition that its optimal policy can be expressed as an index policy. This condition is difficult to verify in complex settings and may fail in nonstationary environments like ours, where index policies are often suboptimal (Ortner et al. 2012).[14] Despite these theoretical limitations, index policies dominate real-world applications due to their computational simplicity and analytical tractability.[15] Given our aim of identifying policies that best describe actual lab behavior without assuming optimality, index policies represent the most plausible candidates for modeling behavior.

When extending index policies originally developed for single-play settings to multiple-play settings, a standard approach is to select multiple arms with the highest index values (Komiyama

[12]Widely cited texts such as Russo et al. (2017) and Lattimore and Szepesvári (2020) provide excellent overviews of these algorithms.

[13]In standard multi-armed bandits, where only one arm is pulled per period and arms are independent, an index policy assigns a real-valued index to each arm using arm-specific statistics and selects the highest-indexed arm (Lattimore and Szepesvári 2020). In contextual bandits, indices may also depend on observed context. For Thompson Sampling, the index is sampled from the posterior distribution of the expected reward for each arm, rather than computed deterministically.

[14]However, UCB-like policies can achieve near-optimal performance under certain conditions, such as abrupt changes at unknown periods (Garivier and Moulines 2011).

[15]For example, see Nguyen-Thanh et al. (2019) for implementation at OpenAI, and He et al. (2020) for application at Taobao.

et al. 2015, Lagrée et al. 2016, Zhou and Tomlin 2018). We follow this convention. A remaining complexity in our setting is that the same arm can be pulled multiple times within a period; this can be done by employing an iterative procedure that sequentially identifies and selects the next-best arm, as detailed in Algorithm 1.

Algorithm 1: Dynamic Trial Allocation Algorithm

Input:
Agent choices: Reward weights $\boldsymbol{\theta}$, learning model specification (RF, molecular features), allocation policy π
Exogenous: Horizon $T = \{2005/01/01, ..., 2015/12/31\}$, capacity constraints $\{n_t\}_{t \in T}$
For *each period* $t \in T$*:*
  Observe available projects $K_t(H_{t-1})$ and train learning model $\widehat{P}_t(\cdot \mid H_{t-1}, c_t)$; `/* For` $t = 2005/01/01$`, history includes Pilot Phase data */`
  Initialize allocation vector $a_t = (0, \ldots, 0)$; `/*` $a_t(i)$ `tracks trials allocated to project` i `*/`
  For $n = 1$ **to** n_t*:*
    For *each project* $i \in K_t(H_{t-1})$*:*
      Compute index value $V_{i,a_t(i)+1,t}$ for the $(a_t(i)+1)$-th trial of project i based on predicted $\widehat{p}_{it}$ from $\widehat{P}_t(\cdot \mid H_{t-1}, c_t)$, $\boldsymbol{\theta}$ and π;
    End
    Select project $i^* = \arg\max_{i \in K_t(H_{t-1})} V_{i,a_t(i)+1,t}$; `/* In case of ties, select a project uniformly at random from the tied projects */`
    Update allocation: $a_t(i^*) \leftarrow a_t(i^*) + 1$;
  End
  Execute trials according to allocation vector a_t;
  Observe outcomes and update history H_t;
End

The models we consider, summarized in Table 2, differ in how they compute the index value V. However, they all include a random noise term ϵ_{it} that follows a standard Gumbel distribution, independently and identically distributed (i.i.d.) across projects within each period. This term captures random, unmodeled fluctuations in the labs' perceived value for each option. Additionally, the Gumbel error transforms the choice into a tractable softmax function and prevents deterministic arm selections, which would otherwise produce a non-smooth likelihood function that is difficult to maximize during estimation.

Among the theoretically motivated models, the Greedy model allocates trials to projects with the highest expected reward (including ϵ_{it}) without exploration. Other models incorporate exploration incentives beyond expected reward maximization. For example, the Explore-Then-Commit model prioritizes new projects entering the choice set $K_t(H_{t-1})$, trying each exactly once before reverting

to greedy allocation. The UCB1 model includes a square root exploration bonus that decreases convexly with the number of trials—new projects receive large bonuses while subsequent trials receive progressively smaller ones.

Table 2 Index Value Specifications Across Behavioral Trial Allocation Models

Model	Index Value Specification V_{ijt} (parameters to be estimated: $\boldsymbol{\theta}, \lambda_1, \lambda_2$)
Theoretically Motivated	
Greedy	$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \epsilon_{it}$
Gittins Index	$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \psi(\cdot)\sqrt{Var(\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t))} + \epsilon_{it}$
Thompson Sampling	$\widehat{x}_{ijt}(\widehat{p}_{it}^{DRAW}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \epsilon_{it}$
Explore-Then-Commit	∞ if $J_i(t-1) = 0$, otherwise $\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \epsilon_{it}$
UCB1	$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \sqrt{\frac{\exp(\lambda_1)}{[J_i(t-1)+j]}} + \epsilon_{it}$
Reduced-Form	
1st-Degree Polynomial	$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \lambda_1[J_i(t-1)+j] + \epsilon_{it}$
2nd-Degree Polynomial	$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \lambda_1[J_i(t-1)+j] + \lambda_2[J_i(t-1)+j]^2 + \epsilon_{it}$
Flexible Variance	$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \lambda_1\sqrt{Var(\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t))} + \epsilon_{it}$
Flex Var+Time Discounting	$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \lambda_1\sqrt{Var(\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t))} - \lambda_2[t - \tau_i(t-1)] + \epsilon_{it}$
Combined	
UCB1+Time Discounting	$\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) + \sqrt{\frac{\exp(\lambda_1)}{J_i(t-1)+j}} - \lambda_2[t - \tau_i(t-1)] + \epsilon_{it}$

Note: Due to the difficulty of computing the exact Gittins (1979) index, Brezzi and Lai (2002)'s approximation is used. $\psi(\cdot)$ is defined as

$$\psi(s) = \begin{cases} \sqrt{s/2} & \text{if } s \leq 0.2 \\ 0.49 - 0.11s^{-1/2} & \text{if } 0.2 < s \leq 1 \\ 0.63 - 0.26s^{-1/2} & \text{if } 1 < s \leq 5 \\ 0.77 - 0.58s^{-1/2} & \text{if } 5 < s \leq 15 \\ \{2log(s) - log(log(s)) - log(16\pi)\}^{-1/2} & \text{if } s > 15, \end{cases}$$

where $s = \frac{\widehat{Var}_{it}}{-ln(\beta)\widehat{p}_{it}(1-\widehat{p}_{it})}$. The discount factor β is set to 0.95. $Var(\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t)) = \int((\widehat{x}_{ijt}(p_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t) - \int \widehat{x}_{ijt}(p_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t)\, d\widehat{P}_t(p_{it} \mid H_{t-1}, c_t))^2\, d\widehat{P}_t(p_{it} \mid H_{t-1}, c_t)$ represents the variance of the predicted reward of trial j of project i based on historical data. $\widehat{p}_{it}^{DRAW}$ in Thompson Sampling is a random draw from the distribution of the predicted probability of success generated by the RF model $\widehat{P}_t(\cdot \mid H_{t-1}, c_t)$. $J_i(t-1)$ represents the number of trials the lab had previously allocated to project i prior to period t, and j represents the jth trial considered on day t. Together, $[J_i(t-1)+j]$ represents the overall order of the trial for project i. Recent implementations of UCB1 adopt a constant value instead of $2\ln(N_{lt})$ in Equation (1) (Lattimore and Szepesvári 2020). In our version of UCB1, we do not assign an infinite value to V_{ijt} when $J_i(t-1)+j = 1$. Doing so would result in an infinite likelihood, which complicates both maximum likelihood estimation and the identification of λ_1, as the latter relies on the distribution of the number of trials across projects. Since the value in the square root must be nonnegative, we use $exp(\lambda_1)$ to ensure the numerator is nonnegative in our unconstrained likelihood maximization routine. $\tau_i(t-1)$ is the period in which the last trial on i was performed, from the perspective of period t. For new projects with no prior trials, $\tau_i(t-1) = t$.

Among the reduced-form models, the polynomial specifications use polynomial terms of trial order to approximate the value of exploring less-tried projects. A negative λ_1 would be consistent with the empirical pattern that such projects received more trials. The Flexible Variance

model allows the variance of the predicted reward $Var(\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t))$ to flexibly influence index values, capturing how uncertainty in reward beliefs affects allocation decisions. The Flexible Variance+Time Discounting model further incorporates time-discounting for older projects, where $[t-\tau_i(t-1)]$ measures the time since the project's last allocation. A positive λ_2 would be consistent with the observed tendency to rarely revisit long-dormant projects.

Our final specification, UCB1+Time Discounting, combines theoretical and empirical features: UCB1's convex exploration bonus that captures diminishing marginal exploration value, plus time-discounting that devalues older projects.

5. Policy Inference Procedure

This section outlines how we estimate the parameters $\boldsymbol{\theta}, \lambda_1, \lambda_2$ for each behavioral model via maximum likelihood estimation (MLE). MLE finds the parameter values that maximize the likelihood of observing the decision sequence in the data $(a_1, \ldots, a_T)$, where each a_t is chosen from available actions $\mathcal{A}_t$ given context c_t and history $H_{t-1} = (c_1, a_1, x_1, \ldots, c_{t-1}, a_{t-1}, x_{t-1})$ (notations defined in Section 4). Models can then be compared based on their fit and predictive performance at these estimated parameters.

5.1. Addressing the Curse of Dimensionality

The curse of dimensionality in this estimation task arises from two sources: (1) The learning process involves hundreds of factors that influence the agent's beliefs about project reward distributions in complex, nonlinear ways. These beliefs change dynamically as parameters vary and must be continuously updated while the estimator searches for the likelihood-maximizing parameter values. (2) Computing the likelihood of observing action a_t requires considering how often a_t would be chosen among all possible allocation vectors in $\mathcal{A}_t$. This is a combinatorial problem: with $|K_t(H_{t-1})|$ typically reaching tens of thousands of projects and capacity constraint n_t averaging over 20 trials per day, the resulting vector space becomes extremely large.

The key simplification addressing (1) is that we have already extracted the component of the labs' beliefs about rewards that requires updating—the predicted trial success probability $\widehat{p}_{it}$.

We replicated the labs' updating process and precomputed $\widehat{p}_{it}$ for each project and period based on observed history. These $\widehat{p}_{it}$ values can be directly incorporated into the reward and index calculations and remain fixed throughout the estimation, even as parameter changes in $\boldsymbol{\theta}, \lambda_1$, and λ_2 cause reward and index values to vary at each iteration. This greatly reduces computational complexity.

To address (2), we introduce Algorithm 2, which operates on a non-combinatorial action space while producing identical allocation patterns as Algorithm 1 under appropriate assumptions about the V_{ijt} functional form.

Algorithm 2: Dynamic Trial Allocation Algorithm (Non-Combinatorial)

Input:
Agent choices: Reward weights $\boldsymbol{\theta}$, learning model specification (RF, molecular features), allocation policy π
Exogenous: Horizon $T = \{2005/01/01, ..., 2015/12/31\}$, capacity constraints $\{n_t\}_{t \in T}$
For *each period* $t \in T$*:*
  Observe available projects $K_t(H_{t-1})$ and train learning model $\widehat{P}_t(\cdot \mid H_{t-1}, c_t)$;
  For *each project* $i \in K_t(H_{t-1})$ *and potential trial* $j \in \{1, 2, \ldots, n_t\}$*:*
    Compute index value V_{ijt} based on predicted $\widehat{p}_{it}$ from $\widehat{P}_t(\cdot \mid H_{t-1}, c_t)$, $\boldsymbol{\theta}$ and π;
  End
  Sort all project-trial pairs (i, j) in descending order of V_{ijt};
  Let $V_{\text{threshold}}$ be the n_t-th largest value of V_{ijt};
  Define allocation set $\boldsymbol{a}_t = \{(i, j) : V_{ijt} > V_{\text{threshold}}\}$; `/* Allocate trials with values above threshold */`
  Let $\tilde{\boldsymbol{a}}_t = \{(i, j) : V_{ijt} = V_{\text{threshold}}\}$; `/* Project-trials at threshold */`
  Let $k = n_t - |\boldsymbol{a}_t|$; `/* Remaining slots to fill */`
  if $k > 0$ **then**
    Randomly select k project-trial pairs from $\tilde{\boldsymbol{a}}_t$ and add to $\boldsymbol{a}_t$;
  end
  Execute all trials in $\boldsymbol{a}_t$;
  Observe outcomes and update history H_t;
End

The equivalence between Algorithm 1 and Algorithm 2 holds if and only if index values preserve trial ordering: for each project i and period t, the index value of an earlier trial (j) is at least as large as that of any later trial ($j' > j$), i.e., $V_{ijt} \geq V_{ij't}$ for all $j < j'$.[16] Otherwise, Algorithm 2 may allocate later trials before earlier ones (e.g., a third trial before the first), which is nonsensical; in contrast, Algorithm 1 always allocates trials sequentially (first trial, then second, then third, etc.).

[16] If index values are equal, trials can be relabeled so that lower-indexed (j) trials are allocated first, ensuring correct ordering.

For our Greedy model, this ordering property holds because, for any period t and project i, the index values V_{ijt} and $V_{ij't}$ differ by only one term—$q_{ijt}(\widehat{p}_{it}|H_{t-1}, c_t) \geq q_{ij't}(\widehat{p}_{it}|H_{t-1}, c_t)$.[17] The property holds for similar reasons for the Gittins Index, Thompson Sampling, Explore-Then-Commit, and UCB1 models (with or without time discounting) described in Section 4.3. For the Flexible Variance and Flexible Variance + Time Discounting models, it holds when $\lambda_1 \geq 0$ (i.e., when the lab values exploring high-variance projects). For polynomial-based models, it holds when $\lambda_1, \lambda_2 \leq 0$ or when $\lambda_1 < 0$ with small positive λ_2 relative to $|\lambda_1|$ (as in our estimates) for moderate values of j (on the order of 10^3 or less).

Given this equivalence, Algorithm 2 circumvents the combinatorial explosion by reducing allocation decisions to simple threshold comparisons: If the threshold project-trial were known and fixed in each period, we would only need to compare each project-trial's index value to this threshold to determine whether the trial is allocated. In practice, the threshold project-trial varies with the model class and parameter values. However, because index values are densely distributed across tens of thousands of projects (large $K_t(H_{t-1})$), it is reasonable to assume that the threshold, as a function of $\boldsymbol{\theta}, \lambda_1, \lambda_2$, changes smoothly and continuously under small parameter perturbations in the maximum-likelihood search. Under this assumption, small changes to individual project-trials' index values will not significantly shift the threshold value, allowing us to treat it as effectively fixed relative to individual project-trials. This, in turn, enables straightforward likelihood construction without combinatorial complexity.

5.2. Constructing the Likelihood Function

The first step in constructing the likelihood function is determining the threshold value $V_{thr,t}$ for each day. Given the precomputed $\widehat{p}_{it}$, $\boldsymbol{\theta}$, and allocation policy π (including λ_1, λ_2 if applicable), we compute V_{ijt} values for all project-trial pairs in $K_t(H_{t-1}) \times \{1, 2, \ldots, n_t\}$. The threshold $V_{thr,t}$ is the n_t-th largest among these values. We can then calculate the likelihood of allocating trial j to project i in period t using Equation (8).

[17] $q_{ijt}(\widehat{p}_{it}|H_{t-1}, c_t) = (1-\widehat{p}_{it})^{m_{i,t}(H_{t-1})+j-1}\widehat{p}_{it} \geq (1-\widehat{p}_{it})^{m_{i,t}(H_{t-1})+j'-1}\widehat{p}_{it} = q_{ij't}(\widehat{p}_{it}|H_{t-1}, c_t)$, with equality only when $\widehat{p}_{it} \in \{0, 1\}$.

$$Pr(a_{ijt}=1;\boldsymbol{\theta},\lambda_1,\lambda_2,H_{t-1},c_t,\pi)=Pr(V_{ijt}>V_{thr,t};\,\boldsymbol{\theta},\lambda_1,\lambda_2,H_{t-1},c_t,\pi). \tag{8}$$

For the Greedy policy, this becomes:

$$\begin{aligned} Pr(a_{ijt}=1;\boldsymbol{\theta},H_{t-1},c_t,\pi) &= Pr(\widehat{x}_{ijt}(\widehat{p}_{it},\boldsymbol{\theta}|H_{t-1},c_t)+\epsilon_{it}>\widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1},c_t)+\epsilon_{thr,t}) \\ &= \frac{exp(\widehat{x}_{ijt}(\widehat{p}_{it},\boldsymbol{\theta}|H_{t-1},c_t))}{exp(\widehat{x}_{ijt}(\widehat{p}_{it},\boldsymbol{\theta}|H_{t-1},c_t))+exp(\widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1},c_t))}. \end{aligned} \tag{9}$$

where the equality follows from i.i.d. Gumbel errors, yielding a smooth likelihood function.[18] Likelihood functions for other models in Section 4.3 are derived analogously; see Appendix C for details. Thus, given the observed action a_{ijt} in the data, its likelihood of occurrence under policy π is:

$$Pr(a_{ijt};\boldsymbol{\theta},\lambda_1,\lambda_2,H_{t-1},c_t,\pi)=\begin{cases} Pr(a_{ijt}=1;\boldsymbol{\theta},\lambda_1,\lambda_2,H_{t-1},c_t,\pi) & \text{if } a_{ijt}=1 \text{ (actually allocated)},\\ 1-Pr(a_{ijt}=1;\boldsymbol{\theta},\lambda_1,\lambda_2,H_{t-1},c_t,\pi) & \text{if } a_{ijt}=0 \text{ (non-allocated)}. \end{cases} \tag{10}$$

The total log-likelihood is obtained by summing the log-likelihoods of the observed actions for all project–trial pairs across the time horizon.[19] We estimate the parameters $\widehat{\boldsymbol{\theta}},\widehat{\lambda_1},\widehat{\lambda_2}$ under policy π by maximizing this total log-likelihood:[20]

$$\widehat{\boldsymbol{\theta}},\widehat{\lambda_1},\widehat{\lambda_2}=\underset{\boldsymbol{\theta},\lambda_1,\lambda_2}{\arg\max}\sum_{t}^{T}\sum_{i\in K_t}\sum_{j=1}^{n_t}\log\left[Pr(a_{ijt};\boldsymbol{\theta},\lambda_1,\lambda_2,H_{t-1},c_t,\pi)\right] \tag{11}$$

The separate identification of these parameters is intuitive, based on revealed preferences and variation in project-trial characteristics across choice sets. Consider, for example, the identification of θ_{biomed} and λ_1 in the UCB1 model. If the lab consistently assigns trials to biomedically important projects—regardless of whether those projects are new or extensively trialed—we infer a relatively high value for the weight on biomedical importance (θ_{biomed}) compared to the exploration bonus term (λ_1). Our likelihood function is structured so that a large positive θ_{biomed} increases the

[18] Ano and Martinez-de Albeniz (2023) also use i.i.d. Gumbel errors to smooth the likelihood.

[19] To avoid numerical issues, any probability value $Pr(a_{ijt};\boldsymbol{\theta},\lambda_1,\lambda_2,H_{t-1},c_t,\pi)$ equal to zero is replaced with 10^{-300} before taking logarithms.

[20] When the total number of project–trial pairs, $\sum_{t=1}^{T}|K_t|\times n_t$, is too large for memory-efficient computation, we evaluate the likelihood on a random subsample rather than the full set of project-trial pairs. This approach yields consistent and asymptotically unbiased parameter estimates as long as the subsample is randomly drawn. In our implementation, since $n_t \ll |K_t|$, the number of allocated trials is small relative to $|K_t|\times n_t$. We therefore include all allocated trials in the likelihood calculation and, for non-allocated trials, randomly sample one observation per project per period.

expected rewards and index values of biomedically important projects relative to the distribution of index values in $K_t(H_{t-1}) \times \{1, 2, ..., n_t\}$, while lowering those of less important projects. This increases the likelihood of allocating trials to biomedically important projects while decreasing it for others, thereby maximizing the likelihood of the observed allocation decisions. Conversely, if the lab frequently trials new projects, regardless of biomedical importance, a higher exploration bonus term (λ_1) relative to the weight on biomedical importance (θ_{biomed}) would maximize the likelihood of the observed allocation decisions. The separate identification of other parameters follows similar logic.

6. Policy Inference Results

Table 3 summarizes how well each model fits the choice data for each lab, measured by in-sample log-likelihood at convergence. While model fit varies across labs, several consistent patterns emerge. First, as expected, more flexible models with additional parameters generally achieve higher log-likelihoods.

Second, UCB-based models consistently outperform both other theoretically motivated models and reduced-form specifications, highlighting the advantage of incorporating a convex decreasing function of trial order. Among the theoretically motivated models, UCB1 often achieves a log-likelihood several times smaller in absolute value despite having only one additional parameter. Among models with a single λ parameter, UCB1's log-likelihood is 57–86% of the 1st-Degree Polynomial model and 32–48% of the Flexible Variance model. Among models with two λ parameters, UCB1+Time Discounting's log-likelihood is 34–48% of the 2nd-Degree Polynomial model and 36–59% of the Flexible Variance + Time Discounting model.

Including a time discounting component further improves fit: comparing Flexible Variance to Flexible Variance + Time Discounting, and UCB1 to UCB1 + Time Discounting, log-likelihood often reduces by more than half in absolute value, suggesting that project recency played a key role in labs' allocation decisions.

Overall, UCB1+Time Discounting consistently delivers the best in-sample fit across all labs, indicating that combining a theoretically grounded exploration term with an empirically motivated reduced-form specification yields the most accurate behavioral model.

Table 3 In-Sample Fit: Log-Likelihood Comparison Across Allocation Models

		Total Log-Likelihood			
Model	No. of λ parameters	JCSG	MCSG	NESG	NYSGRC*
Greedy	0	-1,348,971	-678,188	-652,549	-738,084
Gittins	0	-1,303,780	-556,319	-537,689	-640,445
Thompson Sampling	0	-1,216,347	-1,549,642	-1,096,289	-1,332,608
Explore-Then-Commit	0	-1,678,233	-3,723,545	-3,549,819	-12,213,009
UCB1	1	-532,841	-270,643	-196,923	-338,070
1st-Degree Polynomial	1	-616,074	-463,278	-314,621	-598,133
2nd-Degree Polynomial	2	-615,238	-462,554	-270,591	-590,909
Flexible Variance	1	-1,107,427	-629,079	-613,721	-697,545
Flex Var+Time Discounting	2	-412,705	-339,715	-329,225	-488,325
UCB1+Time Discounting	2	-212,503	-157,909	-118,780	-286,268

Note: Each model is estimated separately for the periods 2005–2008 (the second phase of PSI) and 2009–2015 (the third phase of PSI) for each lab, since the parameters $\boldsymbol{\theta}$ are expected to differ across phases due to the greater emphasis on biomedically important projects during the third phase. The total log-likelihood for each model is the sum of the log-likelihoods from both phases. Estimated parameter values are presented in Appendix Tables D1–D4. *Results for NYSGRC should be interpreted with caution due to data quality concerns.

Table 4 shows how accurately each model predicts the observed trial allocations in the estimation sample, evaluated at the estimated parameters. A well-fitting model should assign high predicted probabilities of allocation to trials that were actually allocated and low probabilities to those that were not. The results confirm that UCB1+Time Discounting performs best. For trials that were actually allocated, the model predicts an average allocation probability of 70–91%, considerably higher than all other models across labs. For non-allocated trials, it predicts an average allocation probability of just 0.1–0.6%, again much lower than competing models.

We next evaluate the out-of-sample predictive performance of each model. Specifically, we estimate model parameters using data from 2005–2006 (the initial years of the second phase of PSI) and 2009–2011 (the initial years of the third phase), and assess each model's ability to predict trial allocations in the subsequent periods—2007–2008 and 2012–2015—which are not used in estimation. Table 5 presents the out-of-sample predicted likelihoods of allocation. As with the in-sample results in Table 4, the UCB1+Time Discounting model again performs best, with predictive accuracy closely mirroring its in-sample performance.

While the estimates tables (Appendix Tables D1–D4) are too large to include in the main text, the results from our best-fitting model, UCB1+Time Discounting, appear reasonable for each lab

Table 4 In-Sample Fit: Predicted Likelihood of Allocation Across Models

	Predicted Likelihood of Allocation							
	Actually Allocated Trials				Non-Allocated Trials			
Model	JCSG	MCSG	NESG	NYSGRC*	JCSG	MCSG	NESG	NYSGRC*
Greedy	0.685	0.642	0.567	0.526	0.114	0.006	0.007	0.011
Gittins	0.637	0.650	0.567	0.537	0.117	0.005	0.006	0.009
Thompson Sampling	0.564	0.445	0.405	0.378	0.099	0.019	0.017	0.027
Explore-Then-Commit	0.711	0.712	0.654	0.615	0.089	0.002	0.003	0.005
UCB1	0.854	0.770	0.762	0.682	0.020	0.002	0.002	0.004
1st-Degree Polynomial	0.834	0.647	0.716	0.540	0.013	0.002	0.001	0.006
2nd-Degree Polynomial	0.835	0.647	0.716	0.542	0.013	0.002	0.002	0.005
Flexible Variance	0.688	0.652	0.572	0.532	0.084	0.004	0.005	0.008
Flex Var+Time Discounting	0.710	0.761	0.682	0.607	0.018	0.002	0.003	0.006
UCB1+Time Discounting	0.909	0.851	0.839	0.701	0.006	0.001	0.001	0.004

Note: Parameter estimates from Appendix Tables D1–D4 are used to predict allocation likelihood, $Pr(a_{ijt} = 1; \widehat{\boldsymbol{\theta}}, \widehat{\lambda_1}, \widehat{\lambda_2}, H_{t-1}, c_t, \pi)$, within the estimation sample. These predicted likelihoods are then averaged separately for the actually allocated trials and the non-allocated trials. *Results for NYSGRC should be interpreted with caution due to data quality concerns.

Table 5 Out-of-Sample Fit: Predicted Likelihood of Allocation Across Models

	Predicted Likelihood of Allocation							
	Actually Allocated Trials				Non-Allocated Trials			
Model	JCSG	MCSG	NESG	NYSGRC*	JCSG	MCSG	NESG	NYSGRC*
Greedy	0.632	0.644	0.552	0.493	0.086	0.008	0.005	0.014
Gittins	0.582	0.653	0.552	0.501	0.093	0.007	0.004	0.011
Thompson Sampling	0.564	0.462	0.406	0.365	0.072	0.020	0.013	0.036
Explore-Then-Commit	0.674	0.692	0.653	0.616	0.067	0.005	0.001	0.003
UCB1	0.833	0.770	0.762	0.707	0.025	0.003	0.001	0.004
1st-Degree Polynomial	0.809	0.704	0.661	0.514	0.012	0.002	0.001	0.006
2nd-Degree Polynomial	0.725	0.704	0.668	0.515	0.016	0.002	0.001	0.006
Flexible Variance	0.670	0.670	0.556	0.504	0.062	0.005	0.003	0.008
Flex Var+Time Discounting	0.717	0.769	0.689	0.602	0.015	0.003	0.002	0.007
UCB1+Time Discounting	0.863	0.866	0.837	0.728	0.008	0.002	0.001	0.003

Note: Each model is estimated separately for 2005–2006 (the initial years of the second phase of PSI) and 2009–2011 (the initial years of the third phase of PSI) for each lab. The resulting estimates are then used to predict allocation likelihood for observations from 2007–2008 and 2012–2015, respectively. Predicted likelihoods are averaged separately for allocated and non-allocated trials. *Results for NYSGRC should be interpreted with caution due to data quality concerns.

and align with our understanding that the labs valued exploration. We find that λ_1 takes on large positive values, indicating a substantial exploration bonus. We also consistently find $\lambda_2 > 0$, suggesting that labs favor more recent projects over older ones. Notably, the estimate $\widehat{\theta}_{\text{biomed}}$ is

higher in the third phase of PSI than in the second across all labs, consistent with the greater emphasis on biomedical importance during that phase—providing a useful validity check.

Several important considerations should be kept in mind when interpreting the fitted models and parameter estimates. First, the parameter estimates are unitless and normalized relative to an error term following a standard Gumbel distribution, so they are not directly comparable across labs (though it remains meaningful to interpret their signs and compare their magnitudes within a lab, provided one assumes that the error-term distribution is consistent across phases within the lab). Second, the estimated $\boldsymbol{\theta}$ weights reflect what and how much the labs valued in their multi-objective decision-making; these are not necessarily aligned with societal values. For example, labs may have prioritized factors that society deems unimportant or undervalued key aspects such as biomedical importance—indeed, during the second phase of PSI, biomedical relevance was underemphasized, sparking debate and leading to its increased prominence in the third phase. Finally, the policy inference exercise does not address whether the best-fitting model represents a better or worse policy in terms of long-term reward maximization. Assessing that would require a simulation-based analysis, which we present in the next section.

7. Simulation-Based Evaluation and Improvement of Decision Policies

In this section, we demonstrate how our policy inference results enable simulation-based evaluation of policy performance for maximizing long-term reward. We simulate trial allocations and rewards under three scenarios of increasing implementation difficulty. The first involves a straightforward algorithmic adjustment requiring no changes to personnel, funding, or equipment—an easily implementable intervention. The second shortens the PSI Pilot Phase, necessitating earlier deployment of specialized personnel and infrastructure for trial data analytics, and thus requiring additional funding and coordination. The third is aspirational: it entails extensive coordination and prolonged data collection to build high-quality resources before the PSI, representing the most challenging but potentially highest-impact intervention. Together, these scenarios illustrate how policy inference can guide incremental evaluation of policy changes and support evidence-based improvement.

To evaluate these interventions, we simulate the sequential interaction between the lab (decision-maker) and nature (environment). This simulation requires two distinct components: **(1) predicting how labs allocate trials** based on trial history, and **(2) modeling how nature generates corresponding rewards.** Our policy inference results address the first component but not the second.

Lab allocation simulation: Our policy inference provides the components needed to simulate lab decisions. We estimated each lab's reward weights, replicated how they updated beliefs about trial success from history, and identified the model class and parameters governing their allocations. Given history H_{t-1}, context c_t, policy π, estimated reward weights $\widehat{\boldsymbol{\theta}}$, and predicted success probabilities $\widehat{p}_{it}$ from the learning model, we can compute index values using the formula for π described in Section 4.3 and simulate allocation decisions by assigning trials with the highest index values. We can also modify these elements—for example, by adopting a more effective learning model that improves $\widehat{p}_{it}$ accuracy or by changing the allocation model class to π'—to simulate counterfactual allocation decisions.

Nature's reward generation: The remaining challenge is modeling nature's mechanism for generating true success probabilities p_{it}. This requires a separate model P_t^*—distinct from labs' learning model $\widehat{P}_t$—since labs' beliefs may not accurately reflect the true data-generating process P_t. This discrepancy arises from several inherent difficulties in recovering true success probabilities from observational data: limited data availability, as labs had only sparse trial histories in earlier years to train their $\widehat{P}_t$ models; data incompleteness, since outcomes for project-trials that were considered but not pursued remain unobserved; and selection bias, because the observed outcomes are unlikely to be randomly distributed.

The implementation of P_t^* resembles $\widehat{P}_t$ in that both use hundreds of molecular properties as predictive features, reflecting the extensive efforts of labs to improve their prediction models. However, P_t^* differs from $\widehat{P}_t$ in key ways to better address the challenge of estimating true success probabilities from observed data. For instance, whereas $\widehat{P}_t$ is a sequence of models updated periodically

as trial history accumulates, P_t^* is a single model trained on the full history H_T, incorporating characteristics and outcomes from all observed trials. In replicating $\widehat{P}_t$, we restrict features to those used in the labs' implementation, but in P_t^* we also include additional variables we know to be important, such as the timing of trials to capture nonstationarity in outcomes. Appendix B details the additional bias reduction measures we implemented to improve P_t^*'s ability to capture the true dynamics of project success probabilities. Our aim is for P_t^* to yield unbiased estimates of these probabilities, which—even if not perfectly accurate—would be sufficient for reliably estimating the long-run rewards required for policy evaluation.

Still, we acknowledge that no model is likely to fully capture nature's reward-generating mechanism in such a complex context. Our use of P_t^* to represent the true data-generating process corresponds to the "direct method" commonly used in the off-policy evaluation literature, in which predicted rewards are generated directly by a model trained on observed data (Farajtabar et al. 2018). We do not employ more advanced techniques, such as importance sampling or doubly robust estimators, because current state-of-the-art techniques generally assume a stationary environment generated the observed data (Dudík et al. 2011, 2014, Farajtabar et al. 2018, Kallus et al. 2022, Si et al. 2023). To our knowledge, the literature has not yet produced reliable and easy-to-implement adaptations of these methods for off-policy evaluation in bandit settings with nonstationary data-generating environments.

The simulation procedure is outlined in Algorithm 3. We clarify a few key points. First, long-term reward and performance under alternative policies are evaluated using the sum of counterfactually simulated rewards across allocated trials: $\sum_{t=t'}^{T'} \sum_{(i,j)\in \boldsymbol{a}'_t} x'_{ijt}$. Second, the reward function weights are set to the estimated values from our best-fitting model (UCB1+Time Discounting) in the Policy Inference exercise, reflecting the lab's revealed preferences for assessing project-trial rewards. Since these weights may not align with societal values, we evaluate performance solely based on inferred lab preferences. If reliable societal weights were available, they could replace lab weights in the reward calculation to assess potential underperformance due to "misaligned" preferences. Third,

we set the new projects entering the counterfactual set of available projects each day to be identical to the actual new projects observed. These projects result from NIH's three mechanisms for project nomination, which we treat as exogenous and do not attempt to model. This also restricts the set of projects to those with observed outcomes—since every NIH-assigned project received at least one trial—mitigating the risk that P_t^* may extrapolate poorly when estimating success probabilities for projects never attempted. Our analysis therefore examines the impact of counterfactual trial allocation across existing projects under alternative policies, rather than the impact of introducing new arms (i.e., projects never assigned by the NIH in reality).

Algorithm 3: Simulation of Sequential Interaction between the Lab and Nature

Input:

Agent choices: Reward weights $\boldsymbol{\theta}'$; `/* set to` $\widehat{\boldsymbol{\theta}}$ `from best-fitting model */`
Learning model specification ; `/* RF, neural net, etc.; which features? */`
Allocation policy π' ; `/* Greedy, Gittins, etc. */`
Pilot phase $\{2000/01/01, ..., t'-1\}$;
Post-pilot period $T' = \{t', ..., 2015/12/31\}$ where t' is transition date;

Exogenous: Actual, full trial history H_T; true success probability model $P_t^*(\cdot \mid H_T, c_T)$;
Capacity constraints $\{n_t\}_{t \in T'}$; ; `/* actual daily number of trials allocated */`
Pilot phase history $H'_{t'-1} = H_{t'-1}$; `/* actual history prior to` t' `*/`
New projects entering $K'_t(H'_{t-1})$ each day ; `/* actual project arrivals */`
Evaluation horizon $\{2005/01/01, ..., 2015/12/31\}$; `/* actual PSI Phase 2 & 3 */`

For *each period* $t \in T'$*:*
  Observe available projects $K'_t(H'_{t-1})$ and train learning model $\widehat{P}'_t(\cdot | H'_{t-1}, c'_t)$;
  For *each potential trial* $(i,j) \in K'_t(H'_{t-1}) \times \{1, ..., n_t\}$*:*
    Compute index value V'_{ijt} based on predicted p'_{it} from $\widehat{P}'_t(\cdot \mid H'_{t-1}, c'_t)$, $\boldsymbol{\theta}'$ and π';
  End
  Sort all project-trial pairs (i,j) in descending order of V'_{ijt};
  Select top n_t trials according to lab capacity constraint;
  For *each allocated trial* (i,j) *in allocation set* $\boldsymbol{a}'_t$*:*
    Generate success probability $p^*_{it} | H'_{t-1}, c'_t, P_t^*(\cdot \mid H_T, c_T)$; `/* true success probability model predicting based on current history and context */`
    Draw trial success outcome $\sim Bernoulli(p^*_{it})$;
    Generate reward $x'_{ijt} = c'_{it} \cdot \boldsymbol{\theta}'$ if this is the first successful trial of i, $x'_{ijt} = 0$ otherwise;
  End
  Update H'_t with new allocations and rewards;
End

Output: Counterfactually simulated long-term rewards $\sum_{t=t'}^{T'} \sum_{(i,j) \in \boldsymbol{a}'_t} x'_{ijt}$

7.1. Alternative Allocation Model Classes

In this set of simulations, we examine whether adopting well-established alternative allocation models can improve the labs' long-term rewards. We focus on models that are ex ante "feasible"—those

Table 6 Simulated Rewards Under Alternative Allocation Models, 2005–2015

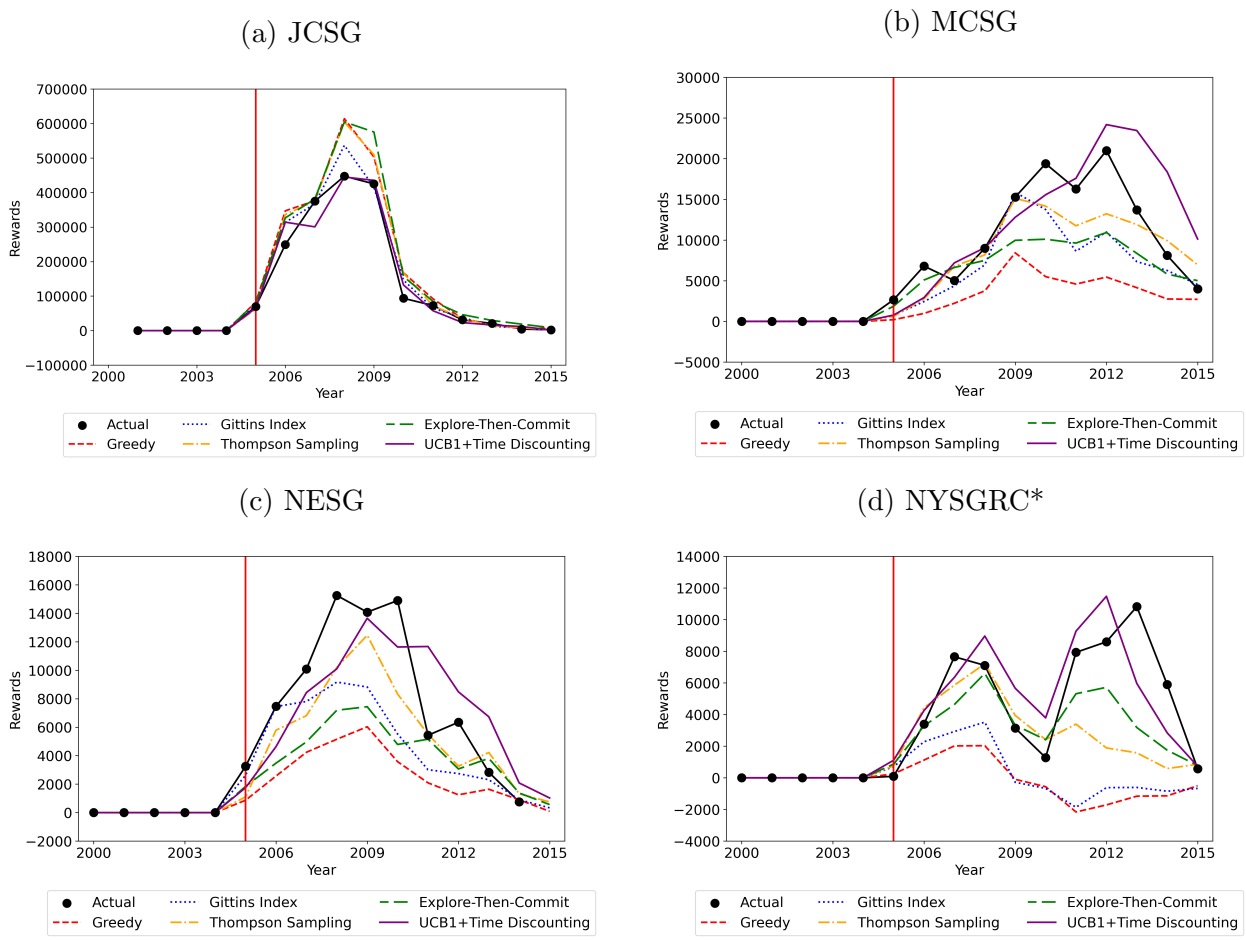
	Total Rewards			
Model	JCSG	MCSG	NESG	NYSGRC*
Greedy	2,245,884	40,729	28,395	-1,932
Gittins	1,976,002	81,841	50,679	3,854
Thompson Sampling	2,186,106	101,402	59,727	32,896
Explore-Then-Commit	2,313,911	80,855	43,663	37,834
Best-fitting model	1,803,448	142,065	80,182	60,405
Actual	1,791,494	121,220	80,385	56,486

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Tables D1–D4. We set $t' = 2005/01/01$, marking the start of the second phase of PSI, when labs adopted the high-throughput approach to trial allocation. For the learning model, we retain the same specification (random forest), features, and hyperparameters used in our replication of the labs' machine learning approach. $\widehat{P'}_t$ is periodically retrained during the simulation as the counterfactual history accumulates. Simulated rewards under the best-fitting model (UCB1+Time Discounting) from the policy inference exercise appear in the second-to-last row. Actual rewards, calculated by summing the observed unique structures' characteristics weighted by $\widehat{\boldsymbol{\theta}}$, are shown in the final row. Negative rewards are possible because we require all available trials to be allocated in each period. Relaxing this constraint would allow labs to skip allocations when expected rewards are negative, resulting in a nonnegative total reward. Additional statistics, including simulated trial distributions, characteristics, and outcomes, are reported in Appendix Tables D5–D8. All reported statistics are averages over three simulation runs per model. *Results for NYSGRC should be interpreted with caution due to data quality concerns.

without free parameters beyond $\boldsymbol{\theta}$—including Greedy, Gittins Index, Thompson Sampling, and Explore-Then-Commit. More sophisticated models with numerous free parameters are challenging to tune without ex post outcome data, making ex ante policy recommendations difficult. In contrast, feasible models can be implemented without extensive domain knowledge and provide practical benchmarks for evaluating current allocation strategies and identifying readily available improvements.

Table 6 reports cumulative simulated rewards, while Figure 2 shows their trajectory over time. If the best-fitting model (UCB1 + Time Discounting) accurately reflects the lab's decision process and our approximation (P_t^*) of the true data-generating process is unbiased, simulated trial allocations, structures, and rewards under this model should closely resemble the actual data. The results in Table 6 and Figure 2 largely support this expectation, with one notable exception: MCSG during 2013–2015, where simulated rewards visually diverge from observed outcomes. Additional statistics and visualizations on trial distributions and structure characteristics are provided in Appendix Tables D5–D8 and Figures D1–D7. These further confirm that simulations using the best-fitting model and P_t^* capture observed patterns reasonably well.

Figure 2 Simulated Reward Trajectories Under Alternative Allocation Models



Note: The period 2000–2004 represents actual historical data (shown to the left of the red vertical line). Because the reward function is not estimated for this period, rewards are displayed as zero. The period from 2005 onward (to the right of the red vertical line) shows simulated annual rewards. Each line represents rewards under a different allocation model, averaged over three simulation runs. For additional plots on simulated trial characteristics and the properties of produced structures, see Appendix Figures D1–D7. *Results for NYSGRC should be interpreted with caution due to data quality concerns.

We next compare alternative allocation policies to the best-fitting model. Results vary by lab. For JCSG, switching to an Explore-Then-Commit policy increases cumulative rewards by up to 28%. For other labs, the best-fitting model remains the highest-performing policy, making straightforward improvements more difficult. Relative policy performance differs across labs, but the Greedy policy is never optimal, highlighting the importance of exploration in this setting.

7.2. Varying the Duration of the PSI Pilot Phase

In this set of simulations, we examine how the duration of the PSI Pilot Phase—when labs relied on ad hoc allocation policies—affects long-term performance. Although little is known about decision-

Table 7 Simulated Rewards (2005–2015) Under Alternative Allocation Models with Shortened Pilot Phase

	Total Rewards				
	JCSG	JCSG	MCSG	NESG	NYSGRC*
Pilot Phase ended by	UCB1+Time Discounting	Explore-Then-Commit	UCB1+Time Discounting	UCB1+Time Discounting	UCB1+Time Discounting
2000	1,946,583	2,519,106	140,077	75,259	60,236
2001	1,933,400	2,587,281	144,312	81,617	60,187
2002	1,894,864	2,413,062	141,106	78,462	60,408
2003	1,877,904	2,319,366	142,352	78,904	59,074
2004	1,860,367	2,277,074	144,105	76,890	61,716
2005 (actual)	1,803,448	2,313,911	142,065	80,182	60,405

Note: Counterfactual simulations examine shortened pilot phases ending by various dates. For the first row, we set $t' = 2000/01/01$, with initial trial history $H'_{t'-1}$ including limited trials from structural biology labs prior to PSI's formal launch. Subsequent rows use t' values of 2001/01/01, 2002/01/01, 2003/01/01, 2004/01/01, and 2005/01/01, respectively. The learning model retains the same specification (random forest), features, and hyperparameters used in our replication of the labs' machine learning approach, with periodic retraining beginning from t' as counterfactual history accumulates. From t' to 2008/12/31, we simulate allocations using the allocation model specified in each column header, with $\widehat{\boldsymbol{\theta}}$ parameters estimated for the best-fitting model (UCB1+Time Discounting) from the 2005–2008 Production Phase (see Appendix Tables D1–D4), effectively starting PSI's second phase at t'. From 2009/01/01 to 2015/12/31, we use $\widehat{\boldsymbol{\theta}}$ parameters estimated for the best-fitting model (UCB1+Time Discounting) from the 2009–2015 Biomedical Phase. The last row reports simulated rewards for 2005–2015 under the actual Pilot Phase (ended by 2005). Additional statistics, including simulated trial distributions, characteristics, and outcomes, are in Appendix Tables D9–D12. See Appendix Figures D8–D15 for visualizations of simulated rewards, trial characteristics, and structure properties over time. All reported statistics are averages over three simulation runs per model. *Results for NYSGRC should be interpreted with caution due to data quality concerns.

making in this phase, we know labs had limited trial data, did not use machine learning, and were not subject to NIH production targets. Because this phase lacked a consistent decision structure, we cannot reliably infer decision policies or simulate extensions of this phase. However, we can simulate scenarios where the Pilot Phase is shortened or eliminated. Unlike switching to a different allocation algorithm in later phases, shortening the Pilot Phase may be more difficult and costly, as it would require labs to hire specialized personnel (e.g., bioinformaticians) earlier to establish machine learning pipelines and enable structured decision making.

In the first scenario, we simulate outcomes assuming no Pilot Phase, with labs adopting machine learning and the best-fitting policy from our inference exercise at the start of PSI in 2000 (for JCSG, we also simulate using the Explore-Then-Commit policy, which outperforms the best-fitting model). We then consider Pilot Phases lasting one (2000), two (2000–2001), three (2000–2002), or four (2000–2003) years.

Table 7 reports cumulative rewards. For JCSG, shorter Pilot Phases consistently improve outcomes. Under UCB1 + Time Discounting, eliminating the Pilot Phase raises 2005–2015 rewards by 8%. Under Explore-Then-Commit, shortening the Pilot to one year raises rewards by 12%. Combined, Explore-Then-Commit with a one-year Pilot yields a 43% increase relative to the best-fitting policy and the actual Pilot Phase duration. Appendix Figures D9–D11 show that earlier adoption of structured policies led JCSG to select different, and potentially more beneficial, trials during 2000–2004.

For other labs, however, shortening the Pilot Phase does not yield clear improvements. As shown in Appendix Figures D9–D11, their 2000–2004 trial characteristics under shortened Pilot scenarios resemble the historical data, suggesting their original Pilot Phase decisions were already similar to later, more structured approaches.

7.3. Improved Information and Learning Models

In this set of simulations, we evaluate how additional training data and improved machine learning models affect trial allocation. In the first scenario, we assume labs have access to the complete observed trial history and know the data-generating process, $P_t^*(\cdot \mid H_T, c_T)$—even before running their own trials. Although unrealistic, this "perfect information" case provides a useful benchmark.

Under perfect information and in the absence of nonstationarity, the Greedy allocation policy without the Gumbel error term (i.e., $V_{ijt} = \widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta} \mid H_{t-1}, c_t)$) should be optimal: one would simply rank trials by expected reward and allocate in descending order. However, with nonstationarity the problem becomes more complex. Allocations must account for time-varying success probabilities, requiring predictions of each trial's success probability on each future date and careful scheduling to maximize long-term reward. In such cases, the Greedy policy without error may no longer be optimal. Computing the true optimal policy remains computationally intensive, so instead we compare two heuristic policies for each lab: the Greedy policy (without error) and the best-performing policy from the first set of simulations (also excluding error).

In subsequent simulations, we reintroduce uncertainty by adding noise to the perfect-information case. These scenarios are more realistic: even with substantial data and a strong understanding of reward dynamics, it is unlikely that labs' beliefs perfectly match the true data-generating

Table 8 Simulated Rewards (2005–2015) Under Improved Information and Learning Models

	Total Rewards			
	JCSG	MCSG	NESG	NYSGRC*
	Greedy	Greedy	Greedy	Greedy
Perfect information	3,201,510	271,166	183,047	183,255
Perfect information, 0.1ϵ	3,208,912	274,479	183,261	174,526
Perfect information, 0.3ϵ	3,226,625	232,669	144,193	100,662
Perfect information, 0.5ϵ	3,202,628	152,950	83,904	75,785
Perfect information, 0.7ϵ	3,167,113	106,942	54,333	63,239
	Explore-Then-Commit	UCB1+Time Discounting	UCB1+Time Discounting	UCB1+Time Discounting
Perfect information	3,228,878	175,215	89,638	90,837
Perfect information, 0.1ϵ	3,241,963	174,495	92,553	90,016
Perfect information, 0.3ϵ	3,209,321	169,961	87,987	88,727
Perfect information, 0.5ϵ	3,195,263	166,883	83,276	87,144
Perfect information, 0.7ϵ	3,209,596	166,671	78,854	82,279
Best-fitting model	1,803,448	142,065	80,182	60,405
Actual	1,791,494	121,220	80,385	56,486

Note: Under perfect information, each lab's belief about a project's success probability equals its true probability (generated by model $P_t^*(\cdot|H_T, c_T)$). Learning models are not separately trained or updated during simulation. The random noise term ϵ follows an i.i.d. standard Gumbel distribution across projects and periods. Additional statistics, including simulated trial distributions, characteristics, and outcomes, are reported in Appendix Tables D13–D16. See Appendix Figures D16–D17 for visualizations of simulated rewards over time. All other notes from Table 6 also apply.

process. To capture this, we reintroduce Gumbel error terms into the allocation model. These represent random, unmodeled fluctuations in perceived project values and capture errors in trial success predictions. By varying the magnitude of these errors, we assess how predictive inaccuracy affects performance. These cases mimic settings where large-scale shared trial history databases improve forecasting but learning remains imperfect. This analysis examines the potential benefits of data-sharing policies and collective learning initiatives, though such efforts require more extensive resources than modifying allocation algorithms or shortening pilot phases.

Table 8 reports cumulative rewards. Under perfect information, the Greedy policy substantially outperforms the previously best-performing models, nearly doubling rewards across major labs by concentrating resources on the highest-value projects. With small predictive errors of 0.1ϵ, similar performance gains are still observed. However, as predictive error increases, the Greedy policy's robustness varies considerably across labs and often deteriorates rapidly, whereas the previously

best-performing policies maintain stable outcomes close to their perfect-information benchmarks. These results suggest that increasing data availability and predictive accuracy can yield substantial gains. Yet, when high accuracy is difficult to achieve, bandit algorithms that explore beyond the apparent best options remain more robust and effective strategies.

8. Discussion and Future Directions

This paper develops a policy inference framework to understand how real-world organizations navigate exploration–exploitation trade-offs in sequential resource allocation. By combining granular choice data with institutional knowledge, we infer the allocation policies most consistent with observed behavior, separately identifying reward functions, learning models, and policy forms. Unlike inverse reinforcement learning, which presumes optimality, or standard policy evaluation, which requires a known policy, our framework uncovers the heuristics organizations actually use and generates counterfactual insights into potential improvements.

Our empirical analysis leverages the NIH-funded Protein Structure Initiative, where research labs conducted sequential trial allocations under extreme uncertainty. This setting provides an unusually rich policy environment: labs operated under shared NIH oversight and standardized reporting requirements, yet exhibited striking heterogeneity in resources and project portfolios. Mapping this environment into a stochastic bandit framework highlights both the relevance of bandit models and the importance of empirically motivated refinements, including correlated outcomes, evolving project sets, and capacity constraints.

Our policy inference procedure recasts the high-dimensional, combinatorial problem of trial allocation into a tractable likelihood-based framework by leveraging two key simplifications: precomputing perceived project-level success probabilities $\widehat{p}_{it}$ to bypass dynamic belief updating, and introducing a threshold-based allocation algorithm that, under mild conditions, is equivalent to the full combinatorial problem. This approach allows us to express choice probabilities in smooth closed form, enabling maximum likelihood estimation of behavioral parameters without sacrificing fidelity to the underlying decision process.

Our policy inference analysis shows that combining theoretical structure with empirically motivated refinements most accurately captures labs' allocation behavior. In particular, UCB1 with time-discounted exploration bonuses consistently provides the best in-sample and out-of-sample fit, outperforming both other theoretically grounded models and reduced-form alternatives. Estimated parameters indicate that labs valued structured exploration, prioritized recent projects, and responded to evolving NIH priorities emphasizing biomedical impact.

Through counterfactual simulations, we demonstrate how inferred policies can guide actionable recommendations. Adopting alternative policies could have yielded meaningful improvements in cumulative rewards, and earlier implementation of structured decision-making could have delivered additional performance gains. We further find that while more accurate reward beliefs enable substantial improvements, purely exploitative strategies, such as Greedy, become fragile even under modest belief errors. In contrast, exploration-based policies remain comparatively robust. These results underscore both the potential and the limits of data-driven improvements in organizational decision-making.

Taken together, our study establishes policy inference as a powerful approach for bridging algorithm design and organizational practice in complex, uncertain environments. Nevertheless, several limitations point to promising avenues for future research. First, our approach relies on highly granular data, which are often difficult to obtain without organizational support. Second, we cannot yet explain why some algorithms perform well for certain decision-makers but not others, due to our limited sample of decision-makers and lack of systematic data on decision-maker characteristics. Third, we do not model how projects are nominated or added to the pool of options (or "arms")—a critical factor that could influence allocation outcomes. Expanding the analysis to broader counterfactual project pools is constrained by our lack of domain expertise in identifying scientifically worthy projects and the absence of reliable models of true reward dynamics for never-attempted projects.

Additional important questions remain beyond the scope of our study, including the optimal configuration of equipment, experimental procedures, and matching of labs to projects. We also

do not assess whether the inferred reward weights were socially optimal. It is possible that using different weights would lead to more numerous or impactful research outcomes, though defining and measuring "more impactful" outcomes remains challenging in this context. In settings with more objective outcome measures—such as pharmaceutical trials with clear revenue or health metrics—such analyses may be more feasible.

Moreover, the set of allocation models we test is not exhaustive. Our framework can only distinguish between policies that produce observably different allocation patterns; if a lab uses a complex algorithm whose choices resemble those of a simpler one, the two cannot be reliably distinguished. Future work can extend our analysis by employing more advanced model selection techniques and testing additional models. Finally, developing robust methods for policy evaluation in nonstationary environments could improve the validity of simulation-based counterfactual analyses.

Despite these limitations, our framework provides an empirical foundation for understanding and improving real-world organizational decision-making under uncertainty. We hope that future work will address these challenges and further advance this understanding.

References

Ano LB, Martinez-de Albeniz V (2023) Inference of a firm's learning process from product launches IESE Business School Working Paper.

Auer P, Cesa-Bianchi N, Fischer P (2002) Finite-time analysis of the multiarmed bandit problem. *Machine Learning* 47(2):235–256.

Babnigg G, Joachimiak A (2010) Predicting protein crystallization propensity from protein sequence. *Journal of Structural and Functional Genomics* 11(1):71–80.

Berman HM, Gabanyi MJ, Kouranov A, Micallef DI, Westbrook J, Protein Structure Initiative network of investigators (2017) Protein Structure Initiative—TargetTrack 2000-2017—all data files. URL `https://zenodo.org/record/821654`, accessed on June 25, 2019.

Brezzi M, Lai TL (2002) Optimal learning and experimentation in bandit problems. *Journal of Economic Dynamics and Control* 27(1):87–108.

Chan AJ, Curth A, van der Schaar M (2022) Inverse online learning: Understanding non-stationary and reactionary policies. *International Conference on Learning Representations*, URL `https://openreview.net/forum?id=DYypjaRdph2`.

Chruszcz M, Wlodawer A, Minor W (2008) Determination of protein structures–A series of fortunate events. *Biophysical Journal* 95(1):1–9.

Dimakopoulou M, Ren Z, Zhou Z (2021) Online multi-armed bandits with adaptive inference. *Advances in Neural Information Processing Systems* 34:1939–1951.

Dudík M, Erhan D, Langford J, Li L (2014) Doubly robust policy evaluation and optimization. *Statistical Science* 485–511.

Dudík M, Langford J, Li L (2011) Doubly robust policy evaluation and learning. *arXiv preprint arXiv:1103.4601* .

Farajtabar M, Chow Y, Ghavamzadeh M (2018) More robust doubly robust off-policy evaluation. *International Conference on Machine Learning*, 1447–1456.

Garivier A, Moulines E (2011) On upper-confidence bound policies for switching bandit problems. *International Conference on Algorithmic Learning Theory*, 174–188.

Gittins JC (1979) Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society: Series B (Methodological)* 41(2):148–164.

He X, An B, Li Y, Chen H, Guo Q, Li X, Wang Z (2020) Contextual user browsing bandits for large-scale online mobile recommendation. *Proceedings of the 14th ACM Conference on Recommender Systems*, 63–72.

Hill R, Stein C (2025) Scooped! Estimating rewards for priority in science. *Journal of Political Economy* 133(3):000–000.

Hortaçsu A, Joo J (2023) *Structural Econometric Modeling in Industrial Organization and Quantitative Marketing: Theory and Applications* (Princeton University Press).

Hotz VJ, Miller RA (1993) Conditional choice probabilities and the estimation of dynamic models. *The Review of Economic Studies* 60(3):497–529.

Hüyük A, Jarrett D, Tekin C, van der Schaar M (2021) Explaining by imitating: Understanding decisions by interpretable policy learning. *International Conference on Learning Representations*, URL `https://openreview.net/forum?id=unI5ucw_Jk`.

Jahandideh S, Jaroszewski L, Godzik A (2014) Improving the chances of successful protein structure determination with a random forest classifier. *Acta Crystallographica Section D: Biological Crystallography* 70(3):627–635.

Jaroszewski L, Slabinski L, Wooley J, Deacon AM, Lesley SA, Wilson IA, Godzik A (2008) Genome pool strategy for structural coverage of protein families. *Structure* 16(11):1659–1667.

Kallus N, Mao X, Wang K, Zhou Z (2022) Doubly robust distributionally robust off-policy evaluation and learning. *International Conference on Machine Learning*, 10598–10632.

Kalvit A, Zeevi A (2021) A closer look at the worst-case behavior of multi-armed bandit algorithms. *Advances in Neural Information Processing Systems* 34:8807–8819.

Kaustov L, Liao J, Lemak S, Duan S, Muhandiram R, Karra M, Srisailam S, Sundstrom M, Weigelt J, Edwards A, Dhe-Paganon S, Arrowsmith C (2007) NMR solution structure of PARC CPH domain. URL `https://www.rcsb.org/structure/2JUF`, accessed on November 1, 2023.

Komiyama J, Honda J, Nakagawa H (2015) Optimal regret analysis of thompson sampling in stochastic multi-armed bandit problem with multiple plays. *International Conference on Machine Learning*, 1152–1161.

Lagrée P, Vernade C, Cappe O (2016) Multiple-play bandits in the position-based model. *Advances in Neural Information Processing Systems* 29.

Lattimore T, Szepesvári C (2020) *Bandit Algorithms* (Cambridge: Cambridge University Press).

March JG (1991) Exploration and exploitation in organizational learning. *Organization science* 2(1):71–87.

McCardle KF, Tsetlin I, Winkler RL (2018) When to abandon a research project and search for a new one. *Operations Research* 66(3):799–813.

Ng AY, Russell S, et al. (2000) Algorithms for inverse reinforcement learning. *International Conference on Machine Learning*, 663–670.

Nguyen-Thanh N, Marinca D, Khawam K, Rohde D, Vasile F, Lohan ES, Martin S, Quadri D (2019) Recommendation system-based upper confidence bound for online advertising. *arXiv preprint arXiv:1909.04190* .

NIGMS (2004) Large-scale centers for the Protein Structure Initiative. URL `https://grants.nih.gov/grants/guide/rfa-files/RFA-GM-05-001.html`, accessed on May 31, 2022.

NIGMS (2008) Protein Structure Initiative (Pilot Phase) fact sheet. URL `http://www.nigms.nih.gov/Initiatives/PSI/Background/PilotFacts.htm`, accessed the Internet Archive capture from Oct 1, 2008.

NIGMS (2009) Concept clearance: High-throughput structural biology URL `https://www.nigms.nih.gov/News/Reports/council_concept_clearance_2009`, accessed the Internet Archive capture from May 23, 2009.

Ortner R, Ryabko D, Auer P, Munos R (2012) Regret bounds for restless Markov bandits. *International Conference on Algorithmic Learning Theory*, 214–228.

Pakes A (1986) Patents as options: Some estimates of the value of holding european patent stocks. *Econometrica* 54(4):755–784.

Petsko GA (2007) An idea whose time has gone. *Genome Biology* 8(6):1–3.

Price WN, Chen Y, Handelman SK, Neely H, Manor P, Karlin R, Nair R, Liu J, Baran M, Everett J, et al. (2009) Understanding the physical properties that control protein crystallization by analysis of large-scale experimental data. *Nature Biotechnology* 27(1):51–57.

Russo D, Van Roy B, Kazerouni A, Osband I, Wen Z (2017) A tutorial on Thompson sampling. *arXiv preprint arXiv:1707.02038* .

Rust J (1987) Optimal replacement of GMC bus engines: An empirical model of Harold Zurcher. *Econometrica* 55(5):999–1033.

Si N, Zhang F, Zhou Z, Blanchet J (2023) Distributionally robust batch contextual bandits. *Management Science* 69(10):5772–5793.

Simchi-Levi D, Wang C (2023) Multi-armed bandit experimental design: Online decision-making and adaptive inference. *International Conference on Artificial Intelligence and Statistics*, 3086–3097.

Slabinski L, Jaroszewski L, Rodrigues AP, Rychlewski L, Wilson IA, Lesley SA, Godzik A (2007a) The challenge of protein structure determination–lessons from structural genomics. *Protein Science* 16(11):2472–2482.

Slabinski L, Jaroszewski L, Rychlewski L, Wilson IA, Lesley SA, Godzik A (2007b) XtalPred: A web server for prediction of protein crystallizability. *Bioinformatics* 23(24):3403–3405.

Thompson WR (1933) On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 25(3/4):285–294.

Van Montfort RL, Workman P (2017) Structure-based drug design: Aiming for a perfect fit. *Essays in Biochemistry* 61(5):431–437.

Varian HR, et al. (2006) Revealed preference. *Samuelsonian economics and the twenty-first century* 99–115.

Wrapp D, Wang N, Corbett KS, Goldsmith JA, Hsieh CL, Abiona O, Graham BS, McLellan JS (2020) Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation. *Science* 367(6483):1260–1263.

Zhou D, Tomlin C (2018) Budget-constrained multi-armed bandits with multiple plays. *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Appendix A: Data and Variable Construction

A.1. Project Rationale/NIH Evaluation Metrics

Several variables capture a lab's observable rationale for allocating a trial to a project, and these variables correspond to the NIH's evaluation metrics for the labs' productivity, among which a key metric is the novelty of the project. The variable $novel_i$ is a binary and is equal to 1 if the labs cited novelty as a reason to allocate trials to project i in the TargetTrack database. Another key NIH evaluation metric is the biomedical importance of the project. The variable $biomed_i$ is a binary and is equal to 1 if the labs cited biomedical importance as a reason to allocate trials to project i in the database.The TargetTrack database contains textual descriptions of why labs allocated trials to a project. The relevant fields are populated for 84% of projects at the four major labs. Construction of $novel_i$ and $biomed_i$ is based on keywords in those descriptions. The following paragraphs describe the variable construction process.

First, we use keywords to identify projects that were novel and/or biomedically important. TargetTrack contains a variable called *targetCategoryList* where labs give projects categorical labels such as "biomedical," "structural coverage,"[21] and so on. It also contains a text field called *targetRationale* where labs give textual descriptions of projects' rationales. Whenever *targetCategoryList* and *targetRationale* contain the following keywords, we set $novel_i$ equal to 1:

big,[22] coverage of protein universe, diversity, first structure of class, low sequence identity, mega,[23] metagenomic, new fold, no structural information, no structure, numer of homologs,[24] pfam, remote homologs, structural coverage, structural template for unsolved, structure coverage, unsolved families, without any solved structures, without structure.

Whenever *targetCategoryList* and *targetRationale* contain the following keywords, we set $biomedical_i$ equal to 1:

[21] "Structural coverage" means the project is in part of the structure space with no or few published structures.

[22] BIG and MEGA domain families were defined by the PSI-2 Target Selection Committee as having high value for extensive coverage. These families contained hundreds to tens of thousands of members and many subfamilies which could not be modeled well due to a lack of structural coverage.

[23] Same as above.

[24] This typo occurs in the raw data.

activator, adhesion, antibiotic, binding, biochemistry, biological interest, biomedical, cascade, catalyze, cell development, community nominated, communit-nominated,[25] community-nominated, community request, conserved, disease, coronavirus, drug, drug development, drug target, effector, enzyme, essential, function, functional studies, functional, gpcr, high value, hig-value,[26] hiv, homeostasis, host, immune, immunity, infection, infectious, inhibitor, interaction, interact, legionella, medical school, metabolism, mitochondria, model system, operon, parkinsons, partnership, pathogen, pathology, pathway, phosphatase, pneumonia, protein family of high biological importance, reagent, receptor, resistance, resistant, salmonella, school of medicine, secret, sensor, shen lab, shen_lab, shen_selection, stem cell, substrate, syndrome, synthesis, t-cell, t cell, therapeutic, thorson lab, toxoplasma, transcription, transport, tuberculosis, tumor, university, vaccine, vibrio, virulence, virulent.

Second, we use labs' selection protocols of projects for additional information. TargetTrack contains a field where labs describe the protocols they used to conduct each stage of the trials. One type of protocol is the selection protocol. For example, 15 projects were selected because of the protocol "TSel_101," which states "These proteins are important for cell development." We read the descriptions associated with each selection protocol and manually classified whether each protocol was "novel" and/or "biomedical."[27] Then we set $novel_i$ equal to 1 if the project was selected due to a "novel" protocol. We set $biomedical_i$ equal to 1 if the project was selected due to a "biomedical" protocol.

Lastly, TargetTrack has a field that contains a list of reference IDs of each molecule in large-scale bioinformatics databases.[28] These reference IDs may yield additional information. Whenever the list of reference IDs contains BIG and MEGA reference IDs,[29] We set $novel_i$ equal to 1.

When the labs cited a project i as being novel, they often emphasized that there were no or few already published structures in the same protein family as i. We therefore construct $prevStruct_{iy}$, a continuous variable (subscripted with the letter y) that captures the number of published structures in the same protein

[25]Same as above.

[26]Same as above.

[27]The manual classification is available upon request.

[28]These reference IDs include, but are not limited to, the molecule's IDs in the Protein Data Bank (PDB), UniProt, and the National Center for Biotechnology Information (NCBI) database.

[29]See footnote 22.

family as i year by year. To construct this variable, we first pull from UniProt the list of protein families $pfam_i$ associated with molecule i. We then obtain a mapping of each protein family to its associated structures from EMBL-EBI (2021) and the structures' publication dates (we take the structure's deposition date to the PDB as the publication date) from Varadi et al. (2020). Merging the datasets results in $prevStruct_{iy}$. If i is associated with multiple protein families, we take the average of the number of already published structures in each protein family associated with i.

As an additional proxy for the biomedical importance of a molecule, we look into the number of publications related to the molecule in UniProt, including structures and other types of publications. We construct $prevPub_{iy}$, a continuous variable that captures the number of publications on molecule i year by year.

Additional NIH evaluation metrics correspond to whether the project was related to human beings, eukaryotes,[30] and the cell membrane. The variable $human_i$ captures how similar molecule i is to any molecules from human beings. When a lab worked on a "human" molecule, often the molecule was actually from bacteria but was very similar to a molecule from human beings and was much easier than the human molecule. Therefore, the right construction for $human_i$ is molecule i's degree of similarity to human molecules rather than being a human molecule itself. We learned this from a conversation with an NIH program officer in charge of the PSI program. To construct this variable, we search each molecule i against all UniProt protein sequences in the Homo sapiens (human) species (UniProt (2021d)). From the search results, we take the maximal percentage identity of i to any human molecule as the variable $human_i$. Due to potentially large number of search results, the search algorithm DIAMOND (Buchfink et al. (2015, 2021)) by default cuts off results at $evalue = 0.001$. $evalue$ is a well-understood metric for search quality in this field. If there are no search results meeting the cutoff, we let $human_i = 0$. For details on how to do the DIAMOND search, please see Appendix A.3.

The variable $eukaryote_i$ likewise captures how similar molecule i is to any molecules from eukaryotes. To construct this variable, we search each molecule i against all UniProt protein sequences in the Eukaryota superkingdom (UniProt (2021c)). From the search results, we take the maximal percentage identity of i to any eukaryotic molecule as the variable $eukaryote_i$. Due to potentially large number of search results, the search algorithm DIAMOND (Buchfink et al. (2015, 2021)) by default cuts off results at $evalue = 0.001$. If there are no search results meeting the cutoff, we let $eukaryote_i = 0$.

[30] See definition in footnote 3.

Table A1 Funding Opportunity Announcements (FOA) Tied To PSI

ID	Title	Year
RFA-GM-99-009	PILOT PROJECTS FOR THE PROTEIN STRUCTURE INITIATIVE	1999
PA-99-116	PROTEIN STRUCTURE INITIATIVE	1999
PA-99-117	PROTEIN STRUCTURE INITIATIVE – SBIR/STTR	1999
RFA-GM-00-006	PILOT PROJECTS FOR THE PROTEIN STRUCTURE INITIATIVE	2000
RFA-GM-05-001	LARGE-SCALE CENTERS FOR THE PROTEIN STRUCTURE INITIATIVE	2004
RFA-GM-05-002	SPECIALIZED CENTERS FOR THE PROTEIN STRUCTURE INITIATIVE	2004
RFA-GM-06-004	Structural Genomics Knowledgebase (U01)	2006
RFA-GM-10-004	PSI:Biology Knowledgebase (U01)	2009
RFA-GM-10-005	Centers for High-Throughput Structure Determination (U54)	2009
RFA-GM-10-006	Centers for Membrane Protein Structure Determination (U54)	2009
RFA-GM-10-007	Consortia for High-Throughput-Enabled Structural Biology Partnerships (U01)	2009
PAR-10-214	High-Throughput-Enabled Structural Biology Research (U01)	2010
PAR-11-176	High-Throughput-Enabled Structural Biology Partnerships (U01)	2011

The variable $membrane_i$ is a binary and is equal to 1 if molecule i is related to the cell membrane. We set this binary variable $= 1$ if project i's UniProt information contains the word "membrane."

A.2. Lab Funding

Funding information comes from two sources. First, the NIH released a series of funding opportunity announcements (FOAs) directly tied to the PSI program (NIH 2019), which allows us to search directly all grants associated with those FOAs on NIH RePORT database (NIH 2021). Table A1 shows the full list of FOAs. Second, labs sometimes received supplementary funds from the NIH, so we also perform a direct search of the labs' names and abbreviations using RePORT's advanced search functionality to obtain data on each labs' supplementary funding. The search term we used was (quotation marks included):

"[lab full name]" OR "[lab abbreviation]"

We then aggregate each lab's sum of research grants by year from the search results.

A.3. Matching Projects to UniProt Molecule Information

As a preliminary to using the UniProt data, we match projects from the TargetTrack database to their molecule information on UniProt through two methods.

TargetTrack has a field containing a list of reference IDs of each molecule i in large-scale bioinformatics databases. These reference IDs include, but are not limited to, the molecule's IDs in the Protein Data Bank (PDB), UniProt, and the National Center for Biotechnology Information (NCBI) database. When the UniProt ID of the molecule is available in this field, the mapping is direct. We also use the following ID

types, which easily convert into UniProt molecule ID through UniProt's ID Mapping service (Huang et al. 2011, UniProt 2021a):

- PDB_ID: a molecule's ID in the Protein Data Bank (PDB), a database for 3D structures.
- P_REFSEQ_AC: a molecule's ID in NCBI's RefSeq protein database.
- EMBL: a molecule's corresponding gene's ID in the European Molecular Biology Laboratory (EMBL)/GenBank/DNA Data Bank of Japan (DDBJ) CDS database.
- P_ENTREZGENEID: a molecule's corresponding gene's ID in GeneID (Entrez Gene) database.
- P_GI: a molecule's GI number assigned by NCBI.

When the first method fails to find a match (usually due to an entirely missing reference ID field or obsolete records in the relevant databases), we use a second method: directly searching the molecule's sequence of amino acids against all protein sequences in UniProt.[31] We perform this search using DIAMOND (Buchfink et al. 2015, 2021), a very fast algorithm for searching similar sequences. The diamond command we used was:

```
diamond blastp -d [database name] -q [input sequences in .fasta]
-o [output in .csv] -f 6 qseqid qlen sseqid slen evalue bitscore pident length
-b4.0 --top 5
```

It produces search results with the following variables:

- *qseqid*: query sequence's identifier (the full sequence in this case).
- *qlen*: query sequence's length.
- *sseqid*: search result's UniProt ID.
- *slen*: search result's length.
- *evalue*: the number of expected hits of similar quality that could be found just by chance in a random database of the same size. E-value is a commonly used measure for the degree of similarity between the query sequence and the search result.
- *bitscore*: the required size of a sequence database in which the current match could be found just by chance. Bit score does not depend on the size of the database and is a common alternative measure for the degree of similarity between the query sequence and the search result.

[31] Downloadable in .fasta format at https://www.uniprot.org/downloads.

- *pident*: percentage of identical matches between the query sequence and the search result over the alignment length.
- *length*: the alignment length between the query sequence and the search result.

If the query sequence's best match search result, determined by the e-value, a standard metric for assessing sequence similarity, has at least 95% *pident* and the alignment length, *length*, is at least 67% of both *qlen* and *slen*, we map the query sequence to the result sequence's UniProt ID.

We were able to match 262,984 (78.4%) of the 335,553 projects to their UniProt entries through the ID mapping method and match an additional 58,593 (17.5%) projects through the direct search. Overall, we were able to map 321,577 (95.8%) projects to their UniProt entries. We then used UniProt's programmatic access for individual entries (UniProt (2021b)) to pull each molecule's information from UniProt. We successfully pulled this information for 319,986 (95.4%) projects.

A.4. Data Glossary

This paper leverages hundreds of project characteristics extracted from diverse sources, primarily used to replicate labs' machine learning models for predicting trial success probabilities and to model how nature generates true trial success probabilities. The following data glossary provides a comprehensive overview of these variables.

* Variable is used as a feature in training $\widehat{P}$, a best-effort replication of the machine learning models used by the labs to predict trial success probabilities.

† Variable is used as a feature in training P^*, a machine learning model representing how nature generates true trial success probabilities.

Please see Appendix B for these models.

Table A2: Data Glossary

Variables	Description

[4 cap letters then 6 digits]$_i$*†	Amino acid attributes from the AAindex database (Kawashima et al. (2007)). Each attribute had an identifier that had four capital letters followed by six digits. We started with the 567 attributes in AAindex1, and then normalized and clustered them to a set of around 30 attribute classes as in Babnigg and Joachimiak (2010). We used scikit-learn's implementation of affinity propagation clustering, which automatically picked 34 clusters. We then kept the cluster center of each class. For each cluster center attribute, we calculated the local average value, the local minimum, and the local maximum of the sum of the attribute in a seven-amino acid sliding window for molecule i as in Babnigg and Joachimiak (2010). This resulted in 102 variables.
[consortium abbreviation]$_{it}$*†	Binary variable $= 1$ if trial j_i was conducted by the given consortium at time t. Only consortia with more than 70 observations of projects in the TargetTrack database have their corresponding variables. 36 variables in total.
[gene]$_i$†	Binary variable $= 1$ if molecule i is coded for by the given gene. From UniProt. We only include genes that have occurred more than 200 times in the data.
[keyword]$_i$†	Binary variable $= 1$ if molecule i is associated with the given keyword in UniProt. Examples of keywords include "Alzheimer disease," "Antioxidant," "RNA-binding," "Viral envelope protein." We only include keywords that have occurred more than 200 times in the data and remove the keyword "3D-structure" because this is the outcome.
[superkingdom-phylum]$_i$*†	Binary variable $= 1$ if molecule i comes from an organism in the specific superkingdom and phylum. From UniProt. Due to the large number of species molecules represented in TargetTrack, we do not go down the UniProt taxonomy below phylum. 81 variables in total.
*aminoAcid*_[X]$_i$*†	Counts the number of times amino acid "X" is in molecule i. 20 variables for each of amino acids A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Contents of certain amino acids are linked to more successes of trials (Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014)).
*aminoAcidPercent*_[X]$_i$*†	Calculate the amino acid "X" content in molecule i in percentages. 20 variables for each of amino acids A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Contents of certain amino acids are linked to more successes of trials (Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014)).
$biomedical_i$*†	Binary variable $= 1$ if project i was biomedically important. See Appendix A.1 for variable construction.
$eukaryote_i$*†	Maximal percentage identity of molecule i to any eukaryotic molecule. To construct this variable, we search each molecule i against all UniProt protein sequences in the Eukaryota superkingdom (UniProt (2021c)). From the search results, we take the maximal percentage identity of i to any eukaryotic molecule as the variable $eukaryote_i$. Due to potentially large number of search results, the search algorithm DIAMOND (Buchfink et al. (2015, 2021)) by default cuts off results at $evalue = 0.001$. $evalue$ is a well-understood metric for search quality in this field. If there are no search results meeting the cutoff, we let $eukaryote_i = 0$.

$exposedAminoAcid\_[X]_i$*†	Counts the number of times amino acid "X" is on the predicted exposed surface of molecule i. 20 variables for each of amino acids A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y. Exposed surface was predicted using the NetSurfP (Klausen et al. (2019)) program with the cutoff of relative solvent accessibility (rsa) > 0.25. Contents of certain amino acids on the exposed surface of the molecule are linked to more successes of trials (Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014)).
$exposedAminoAcidPercent\_[X]_i$*†	Calculates the amino acid "X" content on the predicted exposed surface of molecule i in percentages. 20 variables for each of amino acids A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y. Exposed surface was predicted using the NetSurfP (Klausen et al. (2019)) program with the cutoff of relative solvent accessibility (rsa) > 0.25. Contents of certain amino acids on the exposed surface of the molecule are linked to more successes of trials (Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014)).
$extinctCoeffReduced_i$*†	Molar extinction coefficient of molecule i with reduced cysteines. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Slabinski et al. (2007b) used the extinction coefficient as a feature to predict project success.
$extinctCoeffOxidized_i$*†	Molar extinction coefficient of molecule i with disulfid bridges. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Slabinski et al. (2007b) used the extinction coefficient as a feature to predict project success.
$funding_{ly}$	Total sum of research grants consortium l received from NIH in year y. See Appendix A.2 for variable construction.
$gaps_i$*†	The average number of insertions in molecule i's alignment compared to homologs in UniProt protein sequences. Computed by searching sequence i against UniProt protein sequences using DIAMOND (Buchfink et al. (2015, 2021)). The output variable *gaps* captures this value. Insertions were included as a feature in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Jahandideh et al. (2014).
$gapOpen_i$*†	The average number of insertion openings in the alignment compared to homologs in UniProt protein sequences. Computed by searching sequence i against UniProt protein sequences using DIAMOND (Buchfink et al. (2015, 2021)). The output variable *gapOpen* captures this value. Insertions were included as a feature in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Jahandideh et al. (2014).
$gravyIndex_i$*†	Grand average of hydropathicity index (GRAVY) of molecule i, used to represent the hydrophobicity value of a molecule. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Hydrophobicity is a key determinant of success of trials (Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014)).
$hasPrevSuccess_{ikt}$*†	Binary variable $= 1$ if at least one previous trial on molecule i successfully completed stage k before date t. For †, two versions of the model were trained: one including variables capturing previous successes, failures, publications, and structures in the same protein families, and one excluding them.

$hasPrevFailure_{ikt}$*†	Binary variable = 1 if at least one previous trial on molecule i failed at stage k before date t. For †, two versions of the model were trained: one including variables capturing previous successes, failures, publications, and structures in the same protein families, and one excluding them.
$human_i$*†	Maximal percentage identity of molecule i to any human molecule. To construct this variable, we search each molecule i against all UniProt protein sequences in the Homo sapiens (human) species (UniProt (2021d)). From the search results, we take the maximal percentage identity of i to any human molecule as the variable $human_i$. Due to potentially large number of search results, the search algorithm DIAMOND (Buchfink et al. (2015, 2021)) by default cuts off results at $evalue = 0.001$. $evalue$ is a well-understood metric for search quality in this field. If there are no search results meeting the cutoff, we let $human_i = 0$.
$instabilityIndex_i$*†	Instability index of molecule i, which is an estimate of the stability of the protein in a test tube. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Instability Index was included as a feature in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Jahandideh et al. (2014).
$isoelectricPoint_i$*†	Isoelectric point of molecule i. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Isoelectric point is a key determinant of success of trials (Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014)).
$membrane_i$*†	Binary variable = 1 if project i's UniProt information contains the word "membrane."
$molecularWeight_i$*†	Molecular weight of molecule i, calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module.
$novel_i$*†	Binary variable = 1 if project i was novel. See Appendix A.1 for variable construction.
$p^*_{i,k',t_{k'}}$†	Predicted probability of success for stage k' of a trial for project i, which started in period $t_{k'}$. When predicting the probability of success for stage k, the values of $p^*_{i,k',t_{k'}}$ for all earlier stages ($k' < k$) are used as features.
$percentCoil_i$*†	Predicted percentage of coil secondary structure in molecule i. Predicted using the NetSurfP (Klausen et al. (2019)) program. Secondary structure features were used in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Jahandideh et al. (2014).
$percentCoiledCoil_i$*†	Percentage of coiled-coil regions in molecule i from UniProt. Coiled-coil regions were used in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014).
$percentDisordered_i$*†	Predicted percentage of disordered region in molecule i. Predicted using the NetSurfP (Klausen et al. (2019)) program. Disordered region was used as a feature in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014).

$percentDisorderedUniprot_i$*†	Percentage of disordered region in molecule i from UniProt. Disordered region was used as a feature in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014).
$percentExposed_i$*†	Predicted percentage of amino acids on the exposed surface of molecule i. Exposed surface was predicted using the NetSurfP (Klausen et al. (2019)) program with the cutoff of relative solvent accessibility (rsa) > 0.25. Extent of the exposed surface of the molecule are linked to more successes of trials (Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014)).
$percentHelix_i$*†	Predicted percentage of helix secondary structure in molecule i. Predicted using the NetSurfP (Klausen et al. (2019)) program. Secondary structure features were used in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014).
$percentLowComplexity_i$*†	Predicted percent low-complexity regions in molecule i. Computed using the SEG program (Wootton (1994)). Low-complexity regions were used as features in Slabinski et al. (2007a,b), Jaroszewski et al. (2008).
$percentSignalPeptide_i$*†	Percentage of signal peptide in molecule i. From UniProt. Molecules containing signal peptides have very low chances of success, as stated by Slabinski et al. (2007a,b), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014).
$percentStrand_i$*†	Predicted percentage of strand secondary structure in molecule i. Predicted using the NetSurfP (Klausen et al. (2019)) program. Secondary structure features were used in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014).
$percentTransmembraneHelices_i$*†	Percentage of transmembrane helices in molecule i. From UniProt. Transmembrane helices were used as a feature in Slabinski et al. (2007a,b), Jaroszewski et al. (2008), Price et al. (2009), Babnigg and Joachimiak (2010), Jahandideh et al. (2014) .
$pfam_i$	A list of protein families associated with molecule i, from UniProt (UniProt (2021b)).
$prevPub_{iy}$*†	Number of publications on molecule i by the start of year y, from UniProt (UniProt (2021b)). For †, two versions of the model were trained: one including variables capturing previous successes, failures, publications, and structures in the same protein families, and one excluding them.

$prevStruct_{iy}$*†	Number of already published structures in the same protein families associated with molecule i by the start of year y. To construct this variable, we first pull from UniProt the list of protein families $pfam_i$ associated with molecule i. We then obtain a mapping of each protein family to its associated structures from EMBL-EBI (2021) and the structures' publication dates (we take the structure's deposition date to the PDB as the publication date) from Varadi et al. (2020). Merging the datasets results in $prevStruct_{iy}$. If i is associated with multiple protein families, we take the average of the number of already published structures in each protein family associated with i. For †, two versions of the model were trained: one including variables capturing previous successes, failures, publications, and structures in the same protein families, and one excluding them.
$prevSuccesses_{ikt}$*†	Number of previous trials on molecule i that have successfully completed stage k before date t. For †, two versions of the model were trained: one including variables capturing previous successes, failures, publications, and structures in the same protein families, and one excluding them.
$prevTrials_{ikt}$*†	Number of previous trials on molecule i that have reached stage k before date t. For †, two versions of the model were trained: one including variables capturing previous successes, failures, publications, and structures in the same protein families, and one excluding them.
$refId_i$	A list of reference IDs of molecule i in TargetTrack, used to map i to its information in UniProt.
seq_i	Sequence representation of molecule i's amino acids, unique identifier of project i.
$seqLength_i$*†	The number of amino acids in molecule i.
$simPrevProj_{it}$	The maximal degree of similarity between project i and all previously attempted projects at time t, measured by the bit score (see Appendix A.3 for the definition of bit score). Computed by searching sequence i against all sequences attempted before time t using DIAMOND (Buchfink et al. (2015, 2021)). The maximum of the output variable *bitscore* among research results was used as $simPrevProj_{it}$.
$surfaceRuggedness_i$*†	Surface ruggedness of molecule i, defined by the total accessible surface of molecule i divided by the accessible surface predicted based on molecular mass. The total accessible surface of the molecule i is calculated by summing the predicted absolute solvent accessibility of each amino acid from NetSurfP (Klausen et al. (2019)). The accessible surface predicted based on molecular mass is calculated using the formula $6.3(molecularMass)^{0.73}$ (Miller et al. (1987)). Jahandideh et al. (2014) used this variable as a feature.
$trialId$	The trial ID of a project-trial in the TargetTrack database. Combined with the sequence, this variable uniquely identifies a project-trial in TargetTrack.
$\widehat{Varp}_{it}$	The variance of the predicted probabilities of trial success for project i on day t, calculated across decision trees. See Appendix B for construction.
$Y_{i,trialId,t}$	Binary variable equal to 1 if trial $trialId$ of project i on date t was successful, and 0 if it failed.

$Y_{i,trialId,k,t}$	Binary variable $=1$ if intermediate stage k of trial $trialId$ of project i on date t was successful. $Y_{i,trialId,0,t}=1$ if DNA was successfully cloned. $Y_{i,trialId,1,t}$ is only defined when $Y_{i,trialId,0,t}=1$ and is equal to 1 if protein was successfully expressed. $Y_{i,trialId,2,t}$ is only defined when $Y_{i,trialId,0,t}=1$ and $Y_{i,trialId,1,t}=1$ and is equal to 1 if protein was successfully purified. $Y_{i,trialId,3,t}$ is only defined when $Y_{i,trialId,0,t}, Y_{i,trialId,1,t}, Y_{i,trialId,2,t}=1$ and is equal to 1 if protein was successfully crystalized for X-ray crystallography or prepared for NMR or cryo-EM. $Y_{i,trialId,4,t}$ is only defined when all previous stages were successful and is equal to 1 if the structure was successfully produced and deposited to the Protein Data Bank (PDB) for publication.
$year_{i,trialId,k,t}$†	The year in which stage k of trial $trialId$ of project i started. For NYSGRC, this variable is set to 0 due to data quality issues related to date reporting for this lab.

Appendix B: Training Machine Learning Models to Predict Trial Success Probabilities

This paper employs two models for predicting trial success probability. The first, $\widehat{P}_t$, captures how labs formed beliefs about these probabilities as they accumulated trial outcome data—a model of their learning process. While $\widehat{P}_t$ aims to provide an unbiased estimate of the labs' beliefs, it does not necessarily produce an unbiased estimate of the true probability of trial success. Our implementation of $\widehat{P}_t$ closely follows the machine learning approach described by the labs in their published journal articles.

The second model, P_t^*, represents the true data generating process of trial success probability P_t and is used to simulate counterfactual outcomes. Training P_t^* differs from training $\widehat{P}_t$ because P_t^* needs to produce an unbiased estimate of the true probability of trial success. Consequently, our implementation of P_t^* deviates from $\widehat{P}_t$ in several ways to correct potential biases and improve upon the machine learning models described by the labs.

In this appendix, we first explain the outcome variable used in model training. Then, we detail our implementation of $\widehat{P}_t$ before discussing the ways in which our implementation of P_t^* deviates from it.

B.1. Observed Trial Outcomes

The main observed trial outcome is whether the trial successfully produced a structure. We define a binary variable $Y_{i,trialId,t}$ to denote the success or failure of trial $trialId$ of project i on day t, where $Y_{i,trialId,t}=1$ indicates a successful trial. We also observe intermediate outcomes: the success or failure of individual stages within each trial. A trial progresses through multiple well-defined sequential stages, including cloning the molecule's DNA, purifying the protein, and studying its structure using X-ray crystallography (among other

methods). A trial is considered successful only upon completion of all stages. We define a binary variable $Y_{i,trialId,k,t}$ to indicate the success or failure of stage k of trial $trialId$ of project i on day t, where $k \in 0,1,2,3,4$. The variables are defined as follows: $Y_{i,trialId,0,t} = 1$ if DNA was successfully cloned. $Y_{i,trialId,1,t}$ is defined only when $Y_{i,trialId,0,t} = 1$ and is equal to 1 if the protein was successfully expressed. $Y_{i,trialId,2,t}$ is defined only when $Y_{i,trialId,0,t} = 1$ and $Y_{i,trialId,1,t} = 1$ and is equal to 1 if the protein was successfully purified. $Y_{i,trialId,3,t}$ is defined only when $Y_{i,trialId,0,t} = 1$, $Y_{i,trialId,1,t} = 1$, and $Y_{i,trialId,2,t} = 1$, and is equal to 1 if the protein was successfully prepared for studying its structure (through X-ray crystallography, NMR, or cryo-EM). $Y_{i,trialId,4,t}$ is defined only when all previous stages were successful and is equal to 1 if the structure was successfully produced and deposited to the Protein Data Bank (PDB) for publication.

We predict the success of a trial stage by stage using these intermediate, stage-specific outcomes, rather than using the final trial outcome as the outcome variable for our prediction models. We adopt this approach for two primary reasons: first, the final trial success rate in the dataset is low (1.6%), leading to a significant class imbalance; second, the labs themselves focused on predicting stage-specific outcomes.

B.1.1. Implementation of $\widehat{P}_t$ Our implementation of $\widehat{P}_t$ fits stage-specific models to leverage information from intermediate stage outcomes. Given that each trial proceeds through multiple sequential stages, the overall probability of trial success is modeled as the product of the success probabilities of each stage: $p_{i,trialId,t} = \prod_{k=0}^{4} p_{i,trialId,k,t}$. The intermediate outcomes $Y_{i,trialId,k,t}$ provide valuable information for predicting future trials' success probabilities at stage k ($p_{i,trialId,k,t}$). To replicate each lab's belief updating process regarding success probability at each stage, we train a chronological sequence of machine learning models. Due to computational constraints, we train a model quarterly between 2005 and 2015, rather than daily. For a given quarter $q(t)$ and each stage $k \in \{0,1,2,3,4\}$, the training set $H_{k,q(t)-1}$ comprises project-trial outcomes $Y_{i,trialId,k,t}$ at stage k realized before quarter $q(t)$, along with their characteristics. The project-trial characteristics used as features for model training and predicting labs' beliefs fall into three categories:[32]

- Physicochemical properties of molecule i based on scientific reasoning. These variables were identified by the series of journal articles the labs published and were quite similar across labs and time.
- Other characteristics of project i, for example, novelty, biomedical importance, and the number of prior publications on molecule i.

[32]Please see Appendix A.4 for the full list of variables used.

- Past successes and failures of project i at stage k.

Then, for the given quarter $q(t)$ and each of the stages $k = 0, 1, 2, 3, 4$, we fit a random forest model $\widehat{P}_{k,q(t)}(\cdot \mid H_{k,q(t)-1}, c_t)$ using `RandomForestClassifier` from python package `scikit-learn`. Random forest is an *ensemble*[33] machine learning method. The algorithm constructs a large number of decision trees at training time. Each decision tree is a learning model that aims to find the project-trial characteristics predictive of success/failure in the training set. When it comes to prediction, the trained random forest classifier $\widehat{P}_{k,q(t)}(\cdot \mid H_{k,q(t)-1}, c_t)$ would pool individual trees and average predicted values of $\{\widehat{p}_{ikt}^{(ntree)}\}$ from individual trees as the final output. Jahandideh et al. (2014) set the number of trees in the random forest to 1000, which we also adopted in our previous draft. In this draft, however, we reduce the number to 100 due to computational constraints.

Decision trees and random forests are known for often overfitting without regularization. To avoid overfitting, we regularize by restricting the hyperparameters *max_depth*,[34] *min_samples_leaf*,[35] *max_features*,[36] and *min_samples_split*.[37] We perform model selection with a grid search of the combinations of the four hyperparameters.[38] For each hyperparameter combination, we evaluate the model with five-fold cross validation using `scikit-learn`'s `cross_validate` function. In each iteration of the cross-validation, the function fits a random forest on four out of five cross-validation folds and then computes the cross-validation score by comparing the model's predictions with the actual data from the remaining fold. We use the average log-likelihood (`log_loss` scoring in `scikit-learn`) as the cross-validation scoring method. We choose the hyperparameter combination that maximizes the average log-likelihood in cross-validation.

[33]Ensemble methods use multiple learning models to obtain better predictive performance than could be obtained from any of the constituent learning models alone.

[34]This hyperparameter determines the maximum depth of each decision tree.

[35]This hyperparameter determines the minimum number of observations a node in the decision tree must have before it can be split.

[36]This hyperparameter determines the maximum number of features to consider when looking for the best split.

[37]This hyperparameter determines the minimum number of observations required to split a node.

[38]To reduce computational burden, we do not perform model selection for all $\widehat{P}_{k,q(t)}$. Rather, for each $k = 0, ..., 4$, we construct $H_{k,T}$ using all outcomes at stage k and only perform model selection for $\widehat{P}_{k,T}$ on this full training set. We then use the selected hyperparameters to train the models $\widehat{P}_{k,q(t)}$ where $q(t) = 2005Q1, 2005Q2, ..., 2015Q4$. The set of *max_depth* used in grid search is $[int(log(sample\_size, 2)), 2 \cdot int(log(sample\_size, 2)), 3 \cdot int(log(sample\_size, 2)), 4 \cdot int(log(sample\_size, 2))]$. The set of $min\_samples\_leaf$ used in grid search is $[1, 2, 4]$. The set of $max\_features$ used in grid search is $[0.075, 0.1, 0.2, 0.3, 0.4]$ of the total number of features. The set of *min_samples_split* used in grid search is $[8, 16, 32, 64, 128]$.

After training the models $\widehat{P}_{k,q(t)}$ for $k = 0, 1, 2, 3, 4$ for a given $q(t)$, we predict $\widehat{p}_{it}$ and $\widehat{Varp}_{it}$ for each project-trial in $K_t(H_{t-1}) \times \{1, ..., n_t\}$ on day t as follows. We first collect the predictions $\{\widehat{p}_{ikt}^{(ntree)}\}$ from the 100 individual decision trees in $\widehat{P}_{k,q(t)}(\cdot \mid H_{k,q(t)-1}, c_t)$, and then compute $\widehat{p}_{it}^{(ntree)} = \prod_{k=0}^{4} \widehat{p}_{ikt}^{(ntree)}$. There are 100 values in the set $\{\widehat{p}_{it}^{(ntree)}\}$. We let

$$\widehat{p}_{it} = \bar{p}_{it}^{(ntree)}, \tag{EC.1}$$

$$\widehat{Varp}_{it} = s^2(p_{it}^{(ntree)}) \tag{EC.2}$$

Although our implementation of $\widehat{P}_t$ closely follows the labs' machine learning approaches, it is not an exact replica. Below, we outline the challenges in perfectly replicating the labs' belief formation processes and highlight where our approach aligns with or diverges from theirs.

- We include as features the union of physicochemical properties identified in the labs' published articles (Slabinski et al. 2007a,b, Jaroszewski et al. 2008, Price et al. 2009, Babnigg and Joachimiak 2010, Jahandideh et al. 2014). This unified set, fixed across all labs and time periods in our implementation, minimizes the risk of selection on unobservables. In contrast, the actual feature sets used by the labs varied somewhat across labs and over time. Capturing all these variations is infeasible, as some were likely undocumented in the published literature over the labs' extended operational history.

- The construction of certain feature variables relies on software packages that are frequently updated or have become obsolete. We make every effort to replicate the labs' original methods as closely as possible (see Appendix A.4).

- Our implementation includes past trial outcomes as feature variables, whereas the labs' implementations did not explicitly incorporate these characteristics. However, it is reasonable to assume that researchers would update their beliefs about a project's potential upon observing the success or failure of a trial.

- We use a random forest as the specification of the learning model for all labs and time periods. In contrast, the machine learning models used by the labs in training and prediction varied across labs and over time. It is impossible to capture all of these potential variations throughout the labs' long operational history, as some may not have been documented in the published articles.

- We set the frequency of "updating" and refitting the chronological series of models at a quarterly interval. In contrast, the labs' actual belief-updating frequency is not clearly documented. We chose the quarterly interval because training models at a finer interval, such as daily, would impose significant computational

and storage burdens. Additionally, the day-to-day changes in the history H_t were relatively small. Therefore, to improve computational tractability, we coarsened the frequency of refitting models to a quarterly basis.

• Our model predicts the overall potential for the success of a trial, while the labs' implementations focused on predicting the potential for success at bottleneck stages of a trial. Specifically, for stages where success rates were generally high (such as cloning the DNA), the labs often did not rely explicitly on rigorous methods like supervised machine learning models to form and update beliefs about success probabilities. In contrast, they did rely on such models to predict the potential for success in more challenging stages, such as crystallizing a molecule and studying its structure through X-ray crystallography.

• The output produced by the labs' models may not exactly match $\widehat{p}_{it}$ and $\widehat{Varp}_{it}$. For example, the model in Slabinski et al. (2007b) predicted the probability of success as an intermediate outcome, with the final output being an integer score between 1 and 5, where 1 represented "optimal" and 5 represented "very difficult." The labs' models did not always predict $\widehat{Varp}_{it}$; when they did, the measure typically involved comparing predictions from multiple models side by side (Slabinski et al. 2007a,b, Babnigg and Joachimiak 2010, Jahandideh et al. 2014). It is reasonable to believe that the labs understood that predictions from different models (or submodels of an ensemble model) differed, and they recognized the value in observing how those predictions varied. However, they did not explicitly formulate an additional metric to measure this variation. This approach seems consistent with the idea that the labs used heuristics to guide their exploration of high-variance projects.

B.1.2. Implementation of P_t^* The implementation of P_t^* is almost identical to that of $\widehat{P}_t$ except for a few deviations. First, a new model $\widehat{P}_{k,q(t)}$ (for stages $k = 0, .., 4$) is trained for every quarter $q(t)$ between 2005Q1 and 2015Q4, incorporating new trial outcomes realized in each quarter. In contrast, a single $P_{k,t}^*$ (for stages $k = 0, ..., 4$) is trained only on the full history H_T. H_T covers the characteristics and outcomes of all trials in the sample period. A t subscript is still included for $P_{k,t}^*$ to reflect that the predictions from this model are nonstationary and time-dependent.

Second, P_t^* uses additional covariates to correct the potential bias of $\widehat{P}_t$ in predicting trial success probabilities. The model $\widehat{P}_t$ may be biased in predicting trial success probabilities because it does not account for the propensity of observing a specific stage of a trial. To see this, think about the probability of success of stage 1 of a trial. We observe stage 1 of a trial only if stage 0 of the trial was successful. If the probabilities of success of stages 0 and 1 are positively correlated, then we are more likely to observe stage 1 of trials that

are more likely to succeed in stage 1. Therefore, models trained with the observed data on stage 1 would produce prediction results that are positively biased. Correcting this bias is simple if we assume that the selection into observing a given stage is only based on observable characteristics of trials. In that case, we can use the predicted probabilities of success at previous stages as propensity scores. As such, we include $p^*_{i,k',t_{k'}}$, the predicted probability of success at each earlier stage $k' < k$, as a covariate when we train $P^*_{k,t}$. The labs' published articles offer no discussion about this source of bias, so we do not include these variables in training $\widehat{P}_t$.

To further improve the predictive power of P^*_t, we incorporate project-trial characteristics that the labs did not include in their machine learning models. Specifically, we add keywords and genes associated with molecule i, as well as the year in which stage k began when training each $P^*_{k,t}$ (for stages $k = 0, ..., 4$). Including the stage start year helps capture nonstationarity in success probabilities, reflecting retooling and phase changes in the labs over time, as noted in our conversations with NIH program officers.

Another key difference between P^*_t and $\widehat{P}_t$ lies in whether P^*_t should include covariates capturing previous trial outcomes, publications, and structures within the same protein families. In simulations, all previous trial outcomes are simulated, and thus do not reflect any change in the project's underlying scientific properties. Accordingly, the counterfactual outcome of a trial should not depend on its simulated prior outcomes. However, one could also argue that in practice, labs may learn from prior successes or failures—improving the execution of future trials even if the scientific properties of the molecule remain constant.

To address this ambiguity, we train two versions of P^*_t: one that includes previous outcomes, publications, and structures as features, and one that excludes them. To determine which version better fits each lab, we simulate trial allocations and outcomes as described in Section 7.1, using each lab's estimated allocation policy (UCB1+Time Discounting) and the corresponding parameter estimates reported in Appendix Tables D1–D4. We use the same learning model specification (random forest), features, and hyperparameters as in our replication of the labs' learning models $\widehat{P}_t$.

The underlying logic is that if we simulate using the labs' estimated allocation policies, then a well-specified P^*_t model should reproduce patterns similar to those observed in the actual data. Appendix Table B1 presents simulated trial allocations and outcomes from 2005–2015 alongside the observed patterns. The results indicate that for JCSG and NESG, including previous outcomes, publications, and structures as features in P^*_t yields simulations that more closely match the actual data. In contrast, for MCSG and NYSGRC, the version of

P_t^* that excludes these features performs better. Accordingly, in all simulations reported in Section 7 and Appendix D, we use the inclusive version of P_t^* for JCSG and NESG, and the exclusive version for MCSG and NYSGRC.

Table B1: Simulated Trial Allocations and Outcomes Under Alternative Specifications of P_t^*

	Projects attempted	Trials per project Avg	Trials per project Std	Unique structures	% Biomedical trials	% Biomedical structures	% Novel trials	% Novel structures	Median sequence length trials	Median sequence length structures	Total Rewards
Including previous outcomes, publications, and structures as features											
JCSG	40,881	9.3	19.6	1,837	0.67	0.74	0.92	0.86	211.3	144.0	1,803,448.2
MCSG	77,479	2.0	7.8	2,591	0.50	0.48	0.34	0.25	243.7	200.7	151,289.1
NESG	59,953	2.2	4.6	980	0.37	0.50	0.69	0.61	207.7	127.0	80,181.6
NYSGRC	59,734	2.5	9.9	1,284	0.71	0.44	0.78	0.60	337.3	299.0	51,211.9
Excluding previous outcomes, publications, and structures as features											
JCSG	40,881	9.3	19.7	1,945	0.68	0.75	0.92	0.85	213.7	147.3	1,932,576.3
MCSG	77,483	2.0	8.0	2,481	0.50	0.48	0.33	0.24	243.0	197.3	142,065.2
NESG	59,953	2.2	4.9	913	0.37	0.45	0.70	0.67	206.0	127.5	68,175.1
NYSGRC	59,733	2.5	9.9	1,363	0.71	0.48	0.78	0.63	336.3	308.7	60,405.2
Actual											
JCSG	40,881	9.3	28.4	1,509	0.71	0.67	0.92	0.95	242.0	219.0	1,791,493.8
MCSG	77,200	2.1	8.6	2,203	0.57	0.54	0.29	0.21	277.0	220.0	121,219.6
NESG	59,946	2.2	3.2	1,063	0.32	0.32	0.76	0.81	209.0	134.0	80,384.7
NYSGRC	59,734	2.5	24.6	1,334	0.75	0.49	0.76	0.66	361.0	319.0	56,485.5

Note: Notes from Table 6 apply.

Appendix C: Likelihood Function Specifications

Table C1: Likelihood Function Specifications Across Behavioral Trial Allocation Models

Model π	Likelihood Function Specification $Pr(a_{ijt};\boldsymbol{\theta},\lambda_1,\lambda_2,H_{t-1},c_t,\pi)$
Greedy	$\frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}$, where $\zeta = \widehat{x}_{ijt}(\widehat{p}_{it},\boldsymbol{\theta}\|H_{t-1},c_t)$, $\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}\|H_{t-1},c_t)$.
Gittins Index	$\frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}$, where $\zeta = \widehat{x}_{ijt}(\widehat{p}_{it},\boldsymbol{\theta}\|H_{t-1},c_t)+\psi(\cdot)\sqrt{Var(\widehat{x}_{ijt}(\widehat{p}_{it},\boldsymbol{\theta}\|H_{t-1},c_t)}$, $\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}\|H_{t-1},c_t)+\psi(\cdot)\sqrt{Var(\widehat{x}_{thr,t}(\boldsymbol{\theta}\|H_{t-1},c_t))}$.
Thompson Sampling	$\frac{1}{100}\sum_{DRAW=1}^{100}\frac{exp(\zeta^{DRAW})}{exp(\zeta^{DRAW})+exp(\zeta_{thr}^{DRAW})}$, where $\zeta^{DRAW} = \widehat{x}_{ijt}(\widehat{p}_{it}^{DRAW},\boldsymbol{\theta}\|H_{t-1},c_t)$, $\zeta_{thr}^{DRAW} = \widehat{x}_{thr,t}^{DRAW}(\boldsymbol{\theta}\|H_{t-1},c_t)$. ζ_{thr}^{DRAW} is from the n_t-th largest index value among all project-trials in day t's choice set, computed from the set of draws $\{\widehat{p}_{it}^{DRAW}, i \in K_t(H_t(t-1)\}$ for these projects. A total of 100 such sets of draws were generated.
Explore-Then-Commit	Let N_t^0 be the number of projects without prior trials on day t. If $N_t^0 \geq n_t$: $Pr(a_{ijt}=1\,\|\,\boldsymbol{\theta},H_{t-1}) = \begin{cases}\frac{n_t}{N_t^0}, & \text{for trial } j=1 \text{ of these projects,}\\ 0, & \text{otherwise.}\end{cases}$ If $N_t^0 < n_t$: $Pr(a_{ijt}=1\,\|\,\boldsymbol{\theta},H_{t-1}) = \begin{cases}1, & \text{for trial } j=1 \text{ of these projects,}\\ \frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}, & \text{otherwise,}\end{cases}$ where $\zeta = \widehat{x}_{ijt}(\widehat{p}_{it},\boldsymbol{\theta}\|H_{t-1},c_t)$, $\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}\|H_{t-1},c_t)$, and ζ_{thr} is from the $(n_t - N_t^0)$-th largest index value among the remaining project-trials.

UCB1 $\frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}$, where

$$\zeta = \widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t) + \sqrt{\frac{\exp(\lambda_1)}{[J_i(t-1)+j]}},$$
$$\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1}, c_t) + \sqrt{\frac{\exp(\lambda_1)}{[J_{thr}(t-1)+j_{thr}]}}.$$

1st-Degree Polynomial $\frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}$, where

$$\zeta = \widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t) + \lambda_1[J_i(t-1)+j],$$
$$\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1}, c_t) + \lambda_1[J_{thr}(t-1)+j_{thr}].$$

2nd-Degree Polynomial $\frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}$, where

$$\zeta = \widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t) + \lambda_1[J_i(t-1)+j] + \lambda_2[J_i(t-1)+j]^2,$$
$$\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1}, c_t) + \lambda_1[J_{thr}(t-1)+j_{thr}] + \lambda_2[J_{thr}(t-1)+j_{thr}]^2.$$

Flexible Variance $\frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}$, where

$$\zeta = \widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t) + \lambda_1\sqrt{Var(\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t))},$$
$$\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1}, c_t) + \lambda_1\sqrt{Var(\widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1}, c_t))}.$$

Flex Var+Time Discounting $\frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}$, where

$$\zeta = \widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t) + \lambda_1\sqrt{Var(\widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t))} - \lambda_2[t-\tau_i(t-1)],$$
$$\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1}, c_t) + \lambda_1\sqrt{Var(\widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1}, c_t))} - \lambda_2[t-\tau_{thr}(t-1)].$$

UCB1+Time Discounting $\frac{exp(\zeta)}{exp(\zeta)+exp(\zeta_{thr})}$, where

$$\zeta = \widehat{x}_{ijt}(\widehat{p}_{it}, \boldsymbol{\theta}|H_{t-1}, c_t) + \sqrt{\frac{\exp(\lambda_1)}{J_i(t-1)+j}} - \lambda_2[t-\tau_i(t-1)],$$
$$\zeta_{thr} = \widehat{x}_{thr,t}(\boldsymbol{\theta}|H_{t-1}, c_t) + \sqrt{\frac{\exp(\lambda_1)}{J_{thr}(t-1)+j_{thr}}} - \lambda_2[t-\tau_{thr}(t-1)].$$

Note: For notation definitions, please see Table 2. For Thompson Sampling, since $\widehat{p}_{it}^{DRAW}$ values are drawn from the distribution of predicted probabilities of success generated by the RF model $\widehat{P}_t(\cdot \mid Ht-1, c_t)$, holding everything else fixed, threshold values and the likelihood can still differ across different sets of draws. We compute an average likelihood over 100 such sets. For Explore-Then-Commit, projects without prior trials have infinite V_{ijt} values for their first trial. We set the allocation probability for these first trials to 1 if the number of such projects, N_t^0, is smaller than the daily capacity n_t, and to $\frac{n_t}{N_t^0}$ if $N_t^0 \geq n_t$.

Appendix D: Additional Details on the Simulation Procedure and Additional Results

In Algorithm 4, we provide additional details of the simulation procedure. To simplify the presentation and aid understanding, the version in Algorithm 3 in the main text omits the specifics of outcome simulation. There, the process is described simply as generating a p^*_{it} from our model of the true data-generating process P^*_t and drawing a trial outcome (0/1) from $Bernoulli(p^*_{it})$. In reality, the simulation is more complex: we simulate stage-specific outcomes for each trial using $P^*_{k,t}$, as described in Appendix Section B.1.2. To enable plotting time trends in trial allocation and outcomes, we also simulate the completion dates for each stage of each trial. Algorithm 4 presents the full details of this procedure.

Algorithm 4: Simulation of Sequential Interaction between the Lab and Nature (with Additional Details on Simulating Stage-Specific Outcomes and Date Generation)

Input:

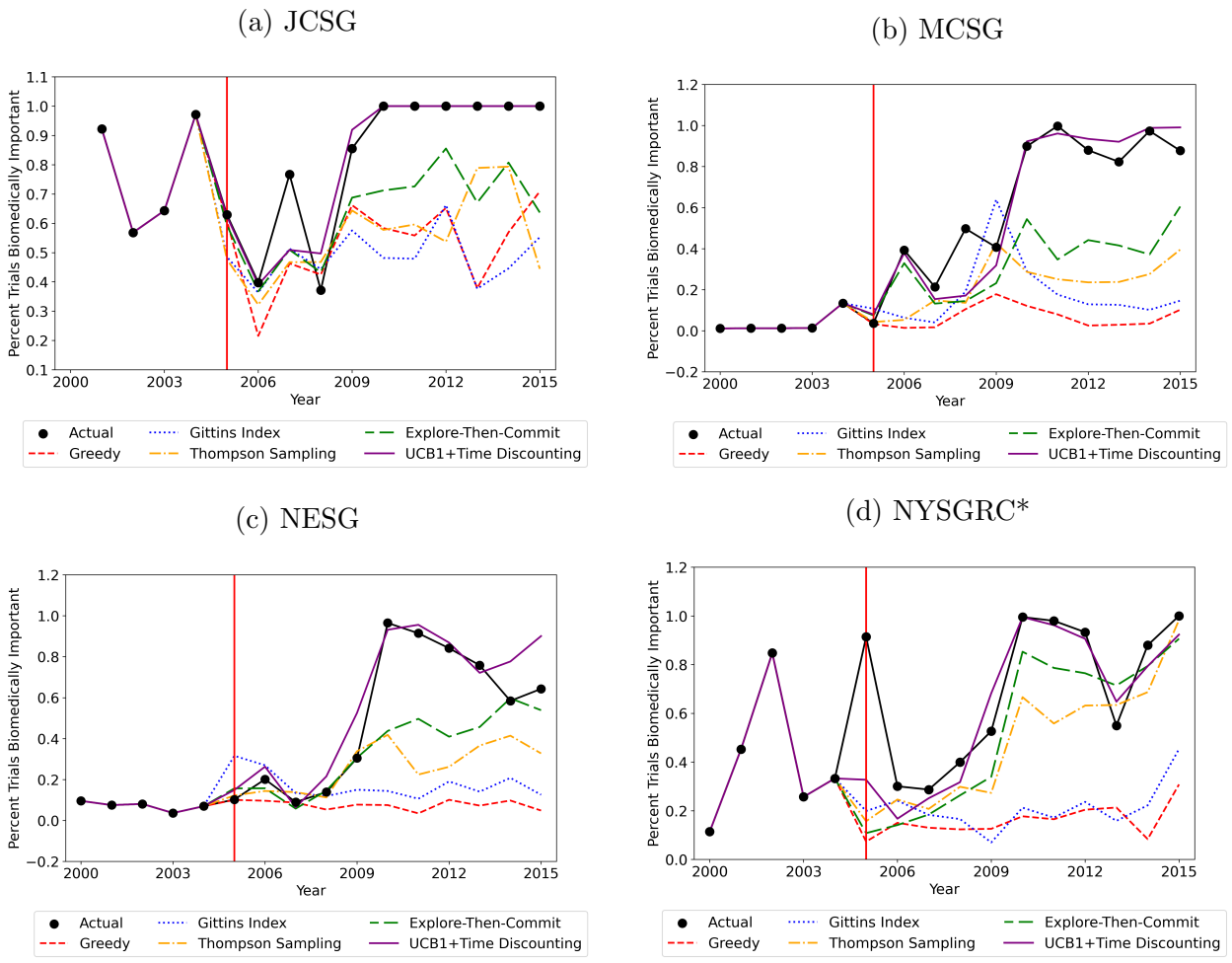
Agent choices: Reward weights $\boldsymbol{\theta}'$; `/* set to` $\widehat{\boldsymbol{\theta}}$ `from best-fitting model */`
Learning model specification ; `/* RF, neural net, etc.; which features? */`
Allocation policy π' ; `/* Greedy, Gittins, etc. */`
Pilot phase $\{2000/01/01, ..., t'-1\}$;
Post-pilot period $T' = \{t', ..., 2015/12/31\}$ where t' is transition date;

Exogenous: Actual, full trial history H_T;
True stage-specific success probability models $P^*_{k,t}(\cdot \mid H_T, c_t)$ for stages $k \in \{0,1,2,3,4\}$;
Capacity constraints $\{n_t\}_{t \in T'}$; ; `/* actual daily number of trials allocated */`
Pilot phase history $H'_{t'-1} = H_{t'-1}$; `/* actual history prior to` t' `*/`
New projects entering $K'_t(H'_{t-1})$ each day ; `/* actual project arrivals */`
Evaluation horizon $\{2005/01/01, ..., 2015/12/31\}$; `/* actual PSI Phase 2 & 3 */`

For *each period* $t \in T'$*:*
  Observe available projects $K'_t(H'_{t-1})$ and train learning model $\widehat{P}'_t(\cdot | H'_{t-1}, c'_t)$;
  For *each potential trial* $(i,j) \in K'_t(H'_{t-1}) \times \{1, ..., n_t\}$*:*
    Compute index value V'_{ijt} based on predicted p'_{it} from $\widehat{P}'_t(\cdot \mid H'_{t-1}, c'_t)$, $\boldsymbol{\theta}'$ and π';
  End
  Sort all trials in descending order of $\widehat{V}'_{ijt}$;
  Select top n_t trials according to lab capacity constraint;
  For *each allocated trial* (i,j) *in allocation set* $\boldsymbol{a}'_t$*:*
    Set stage 0 start date: $t^* = t$;
    For *stage* $k = 0$ *to* 4*:*
      Generate success probability $p^*_{i,k,t^*} | H'_{t-1}, c'_t, P^*_{k,t}(\cdot \mid H_T, c_T)$; `/* true success probability model predicting based on current history and context */`
      Draw stage duration Δt_k from empirical distribution ; `/* Lab-specific empirical distribution of stage` k `durations */`
      Draw trial success outcome $\sim Bernoulli(p^*_{i,k,t^*})$;
      if *outcome = 0 (failure)* **then**
        Terminate trial;
        break;
      end
      Update stage start: $t^* = t^* + \Delta t_k$;
    End
    Generate reward $x'_{ijt} = c'_{it} \cdot \widehat{\boldsymbol{\theta}}$ if this is the first successful trial of i, $x'_{ijt} = 0$ otherwise;
  End
  Update H'_t with new allocations and rewards;
End

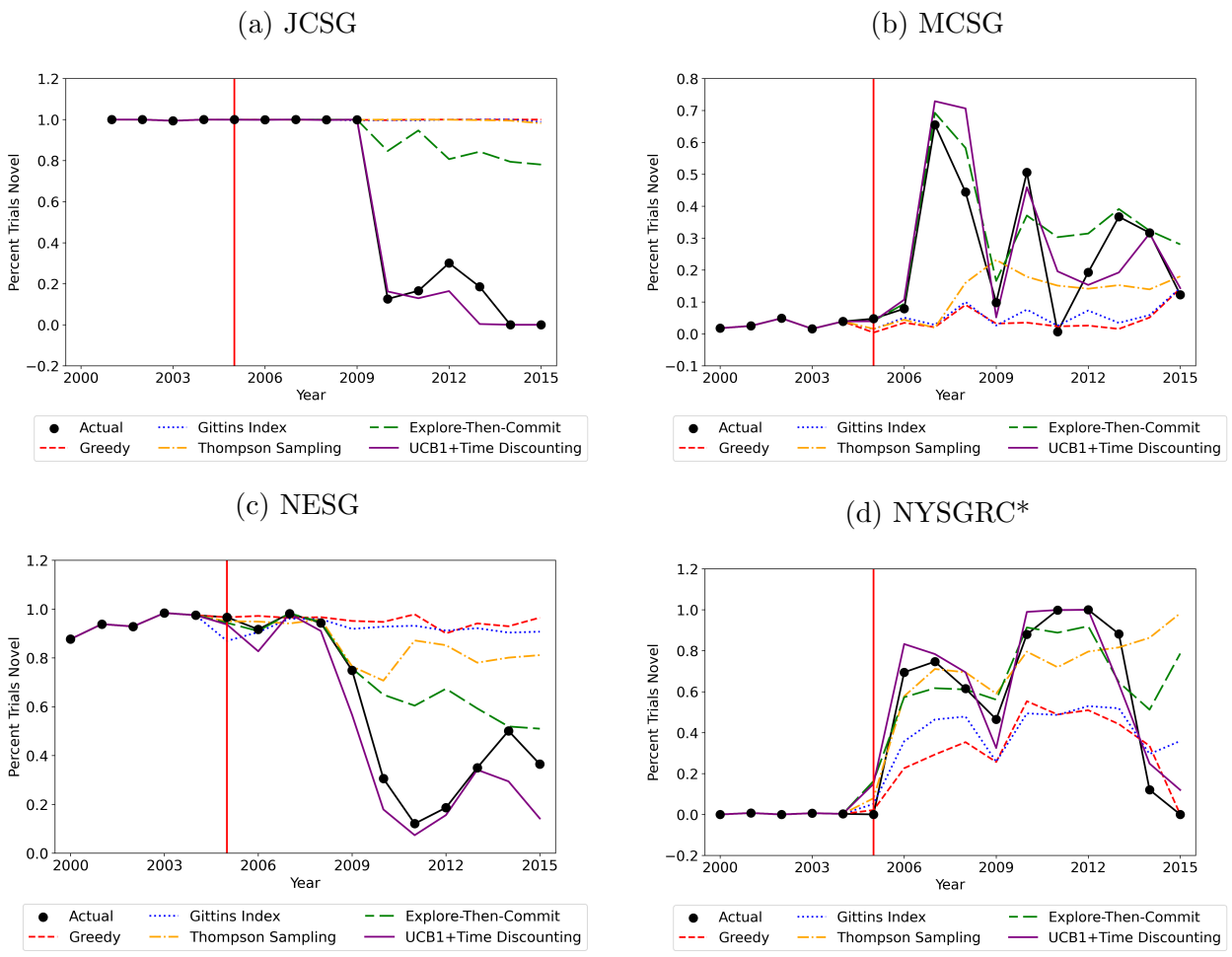
Output: Counterfactually simulated long-term rewards $\sum_{t=t'}^{T'} \sum_{(i,j) \in \boldsymbol{a}'_t} x'_{ijt}$

Figure D1 Proportion of Biomedically Important Trials Under Different Allocation Models Over Time



Note: See notes from Figure 2 for additional details.

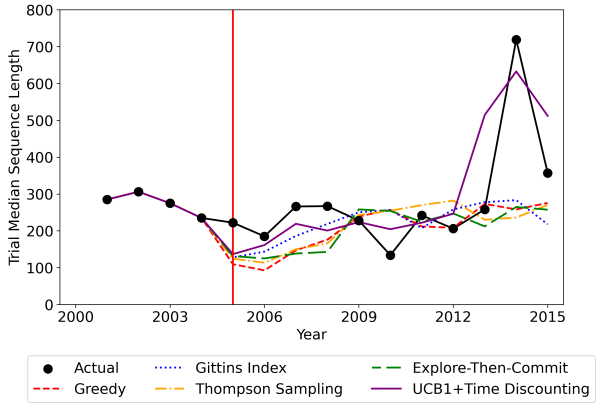
Figure D2 Proportion of Novel Trials Under Different Allocation Models Over Time



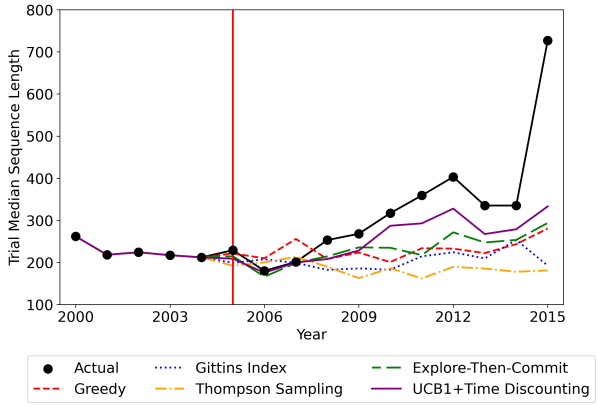
Note: See notes from Figure 2 for additional details.

Figure D3 Trial Median Sequence Length Under Different Allocation Models Over Time

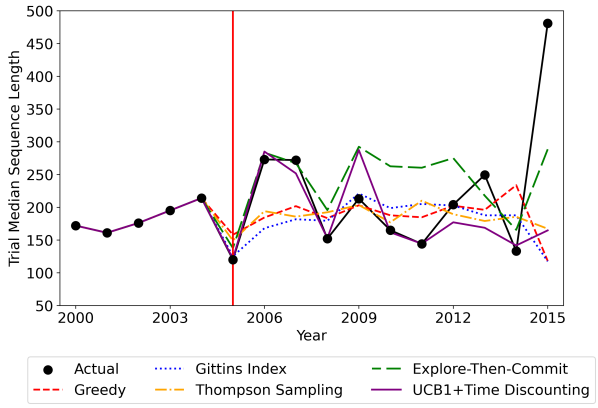
(a) JCSG



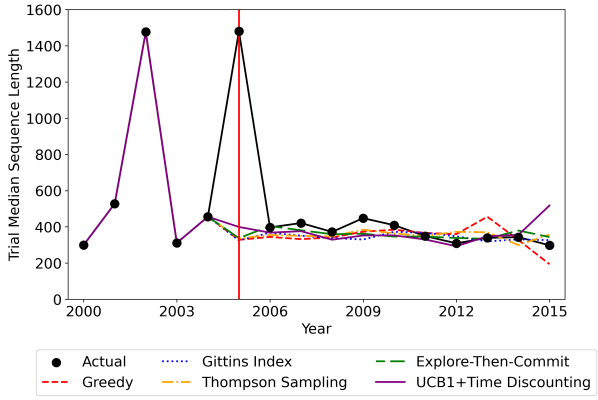
(b) MCSG



(c) NESG

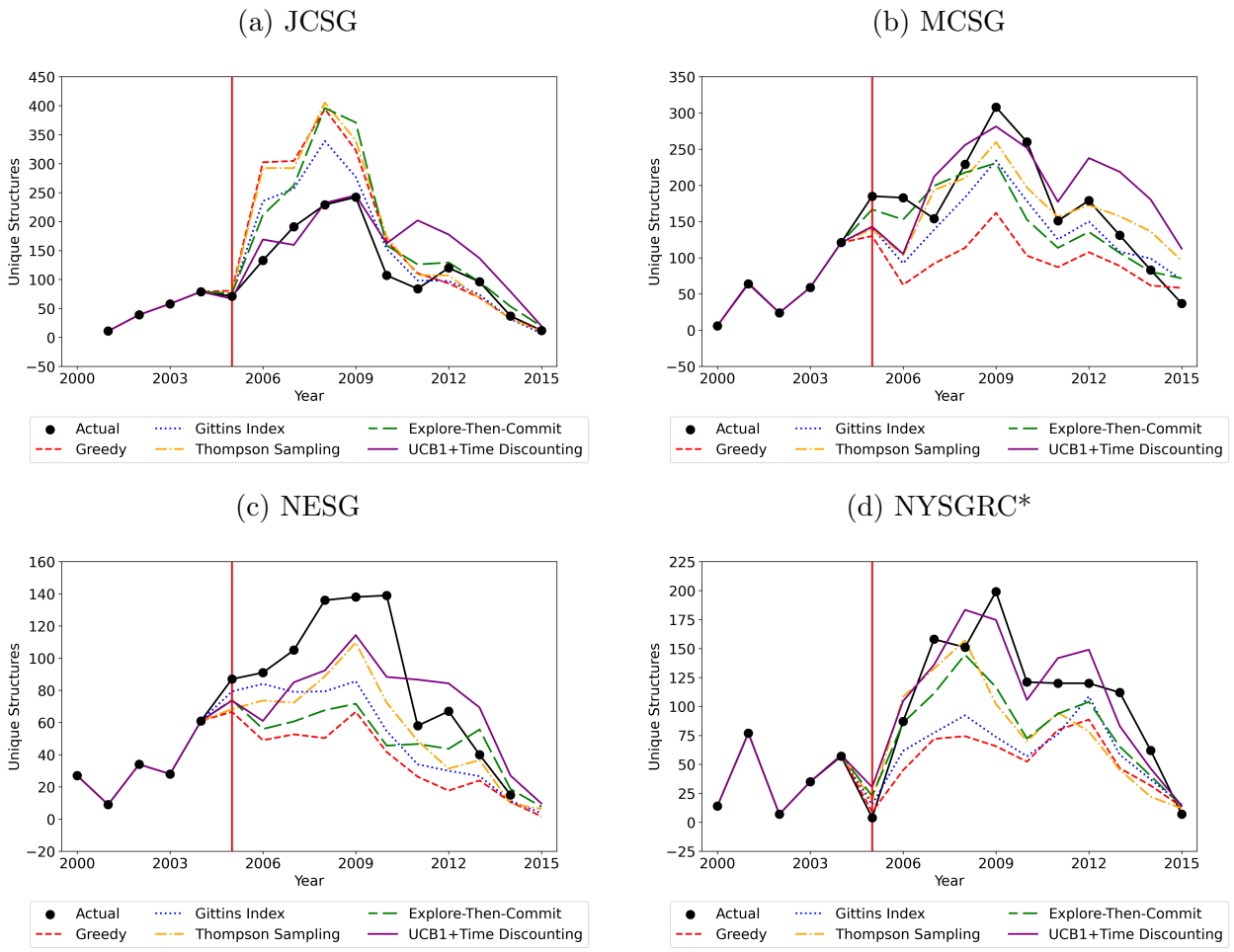


(d) NYSGRC*



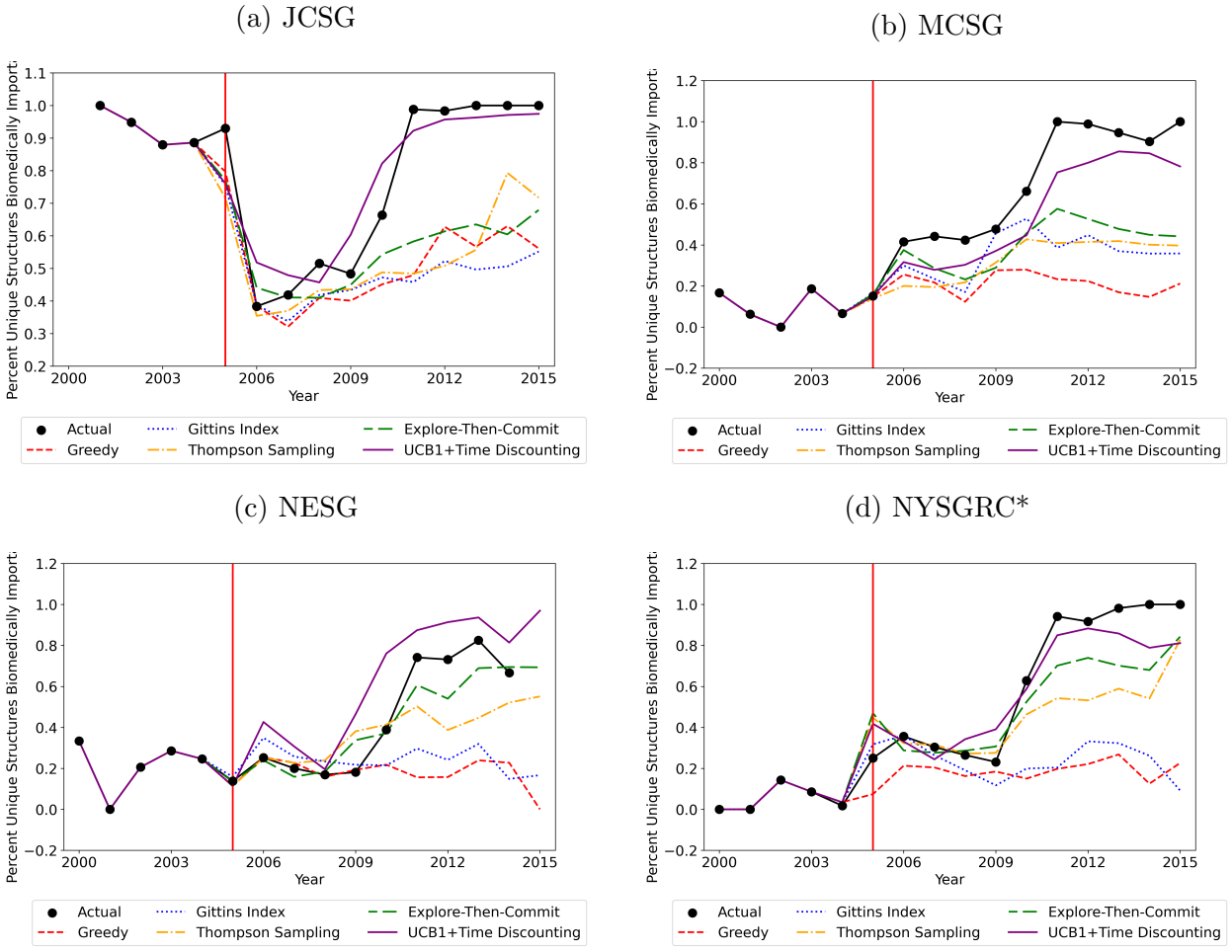
Note: See notes from Figure 2 for additional details.

Figure D4 Number of Unique Structures Produced Under Different Allocation Models Over Time



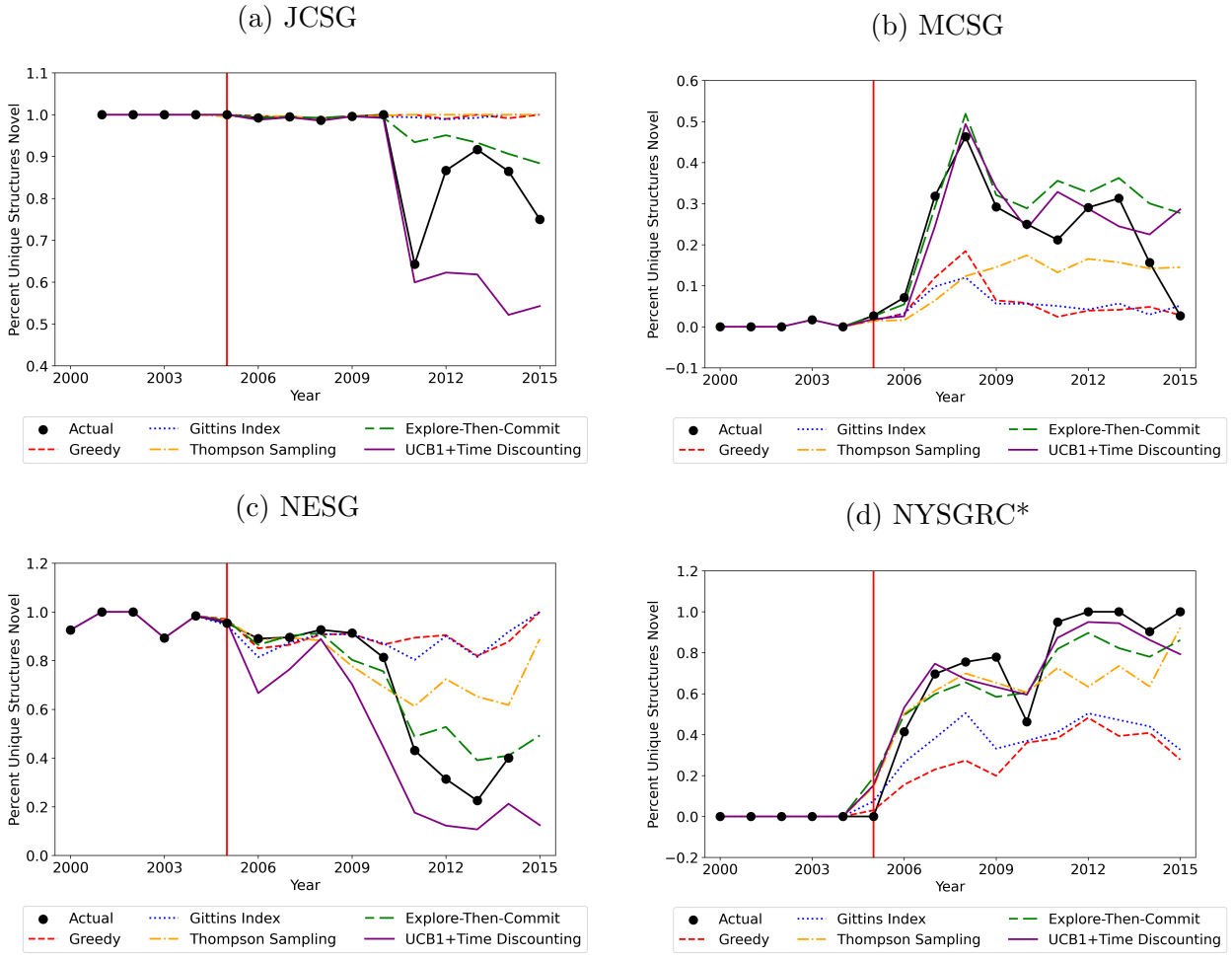
Note: See notes from Figure 2 for additional details.

Figure D5 Proportion of Biomedically Important Structures Under Different Allocation Models Over Time



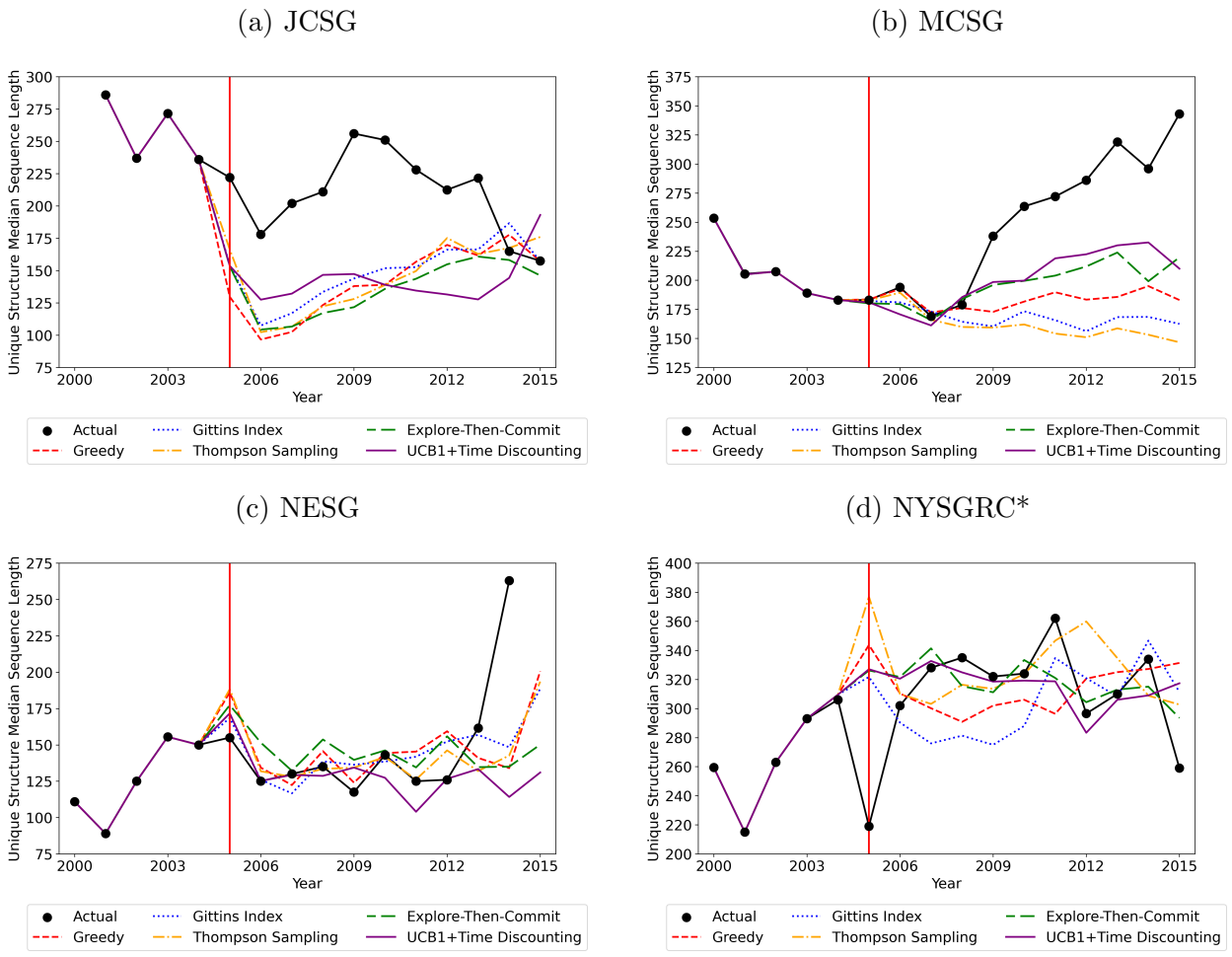
Note: See notes from Figure 2 for additional details.

Figure D6 Proportion of Novel Unique Structures Under Different Allocation Models Over Time



Note: See notes from Figure 2 for additional details.

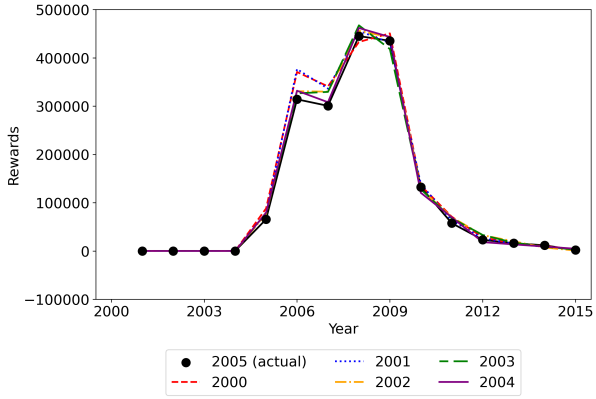
Figure D7 Unique Structure Median Sequence Length Under Different Allocation Models Over Time



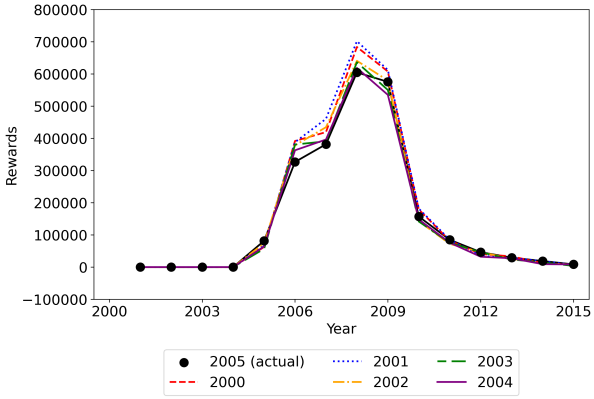
Note: See notes from Figure 2 for additional details.

Figure D8 Simulated Reward Trajectories Under Alternative PSI Pilot Phase Duration

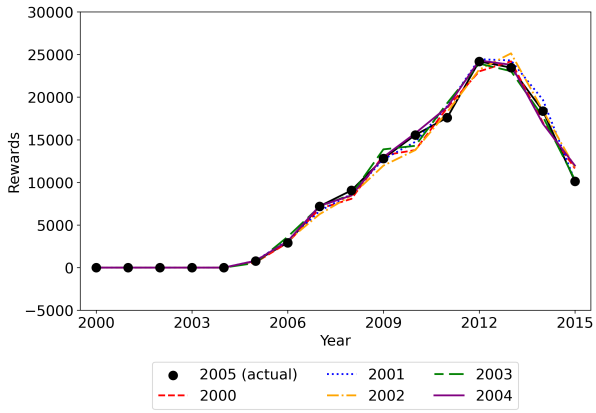
(a) JCSG, UCB1+Time Discounting



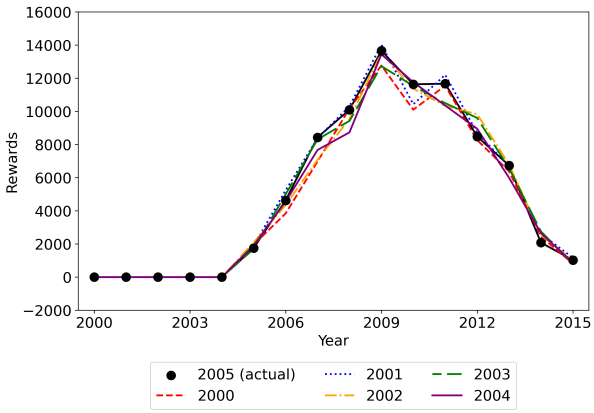
(b) JCSG, Explore-Then-Commit



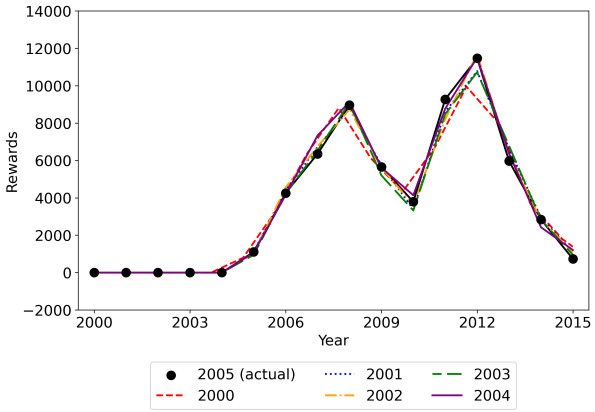
(c) MCSG, UCB1+Time Discounting



(d) NESG, UCB1+Time Discounting



(e) NYSGRC*, UCB1+Time Discounting

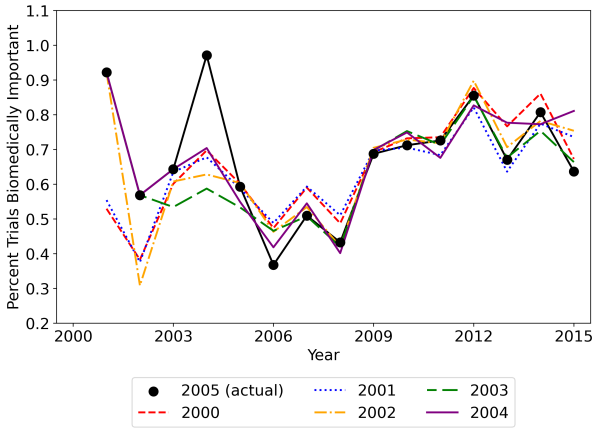


Note: Each line represents simulations with different starting points. The "2005 (actual)" series begins in 2005, the actual end of the Pilot Phase. The "2004," "2003," and earlier series reflect counterfactual scenarios where the Pilot Phase ends earlier, with simulations beginning in those respective years. All simulations use the allocation model specified in the corresponding subfigure title. Values shown are annual rewards averaged across three simulation runs for each scenario. Because the reward function is not estimated for 2000–2004, rewards are displayed as zero for this period.

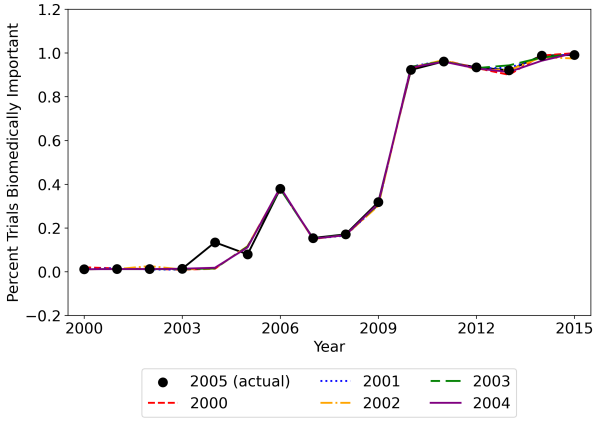
Figure D9 Proportion of Biomedically Important Trials Under Alternative PSI Pilot Phase Duration

(a) JCSG, UCB1+Time Discounting

(b) JCSG, Explore-Then-Commit



(c) MCSG, UCB1+Time Discounting



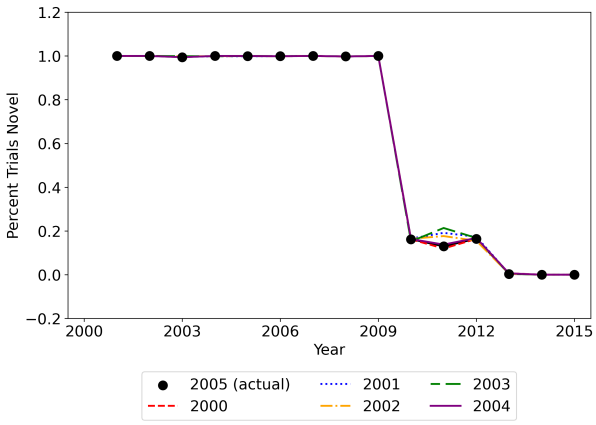
(d) NESG, UCB1+Time Discounting

(e) NYSGRC*, UCB1+Time Discounting

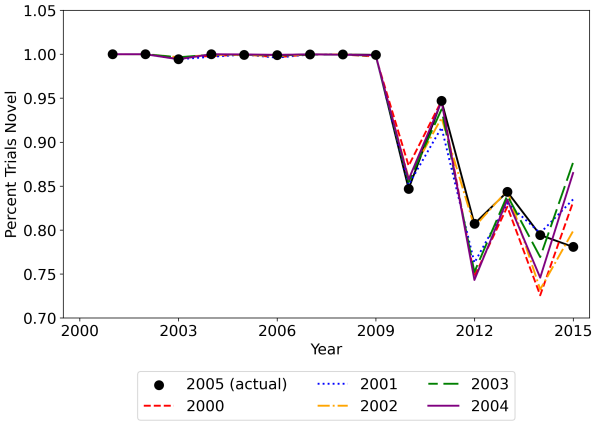
Note: See notes from Appendix Figure D8 for additional details.

Figure D10 Proportion of Novel Trials Under Alternative PSI Pilot Phase Duration

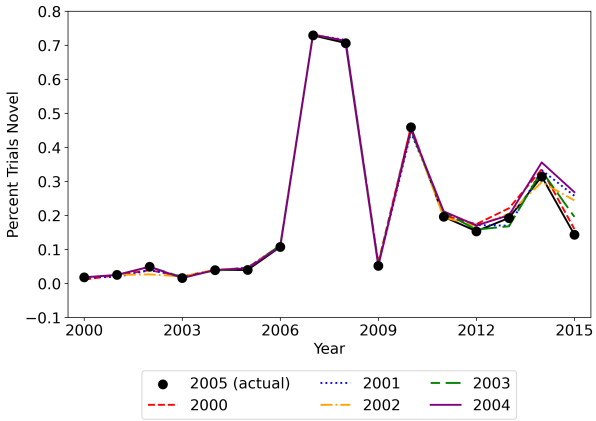
(a) JCSG, UCB1+Time Discounting



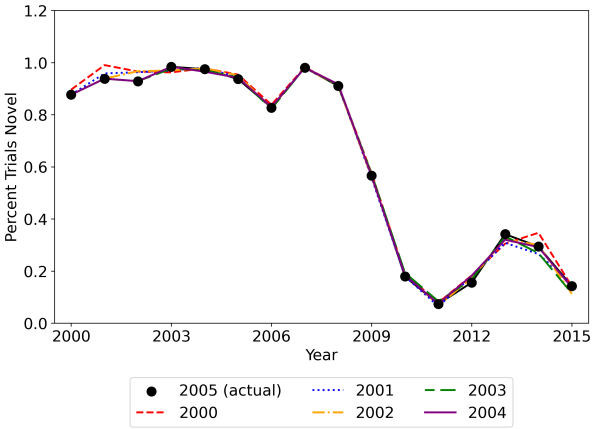
(b) JCSG, Explore-Then-Commit



(c) MCSG, UCB1+Time Discounting



(d) NESG, UCB1+Time Discounting

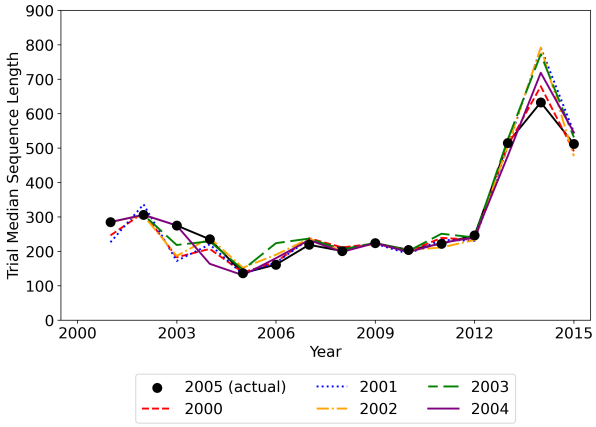


(e) NYSGRC*, UCB1+Time Discounting

Note: See notes from Appendix Figure D8 for additional details.

Figure D11 Trial Median Sequence Length Under Alternative PSI Pilot Phase Duration

(a) JCSG, UCB1+Time Discounting

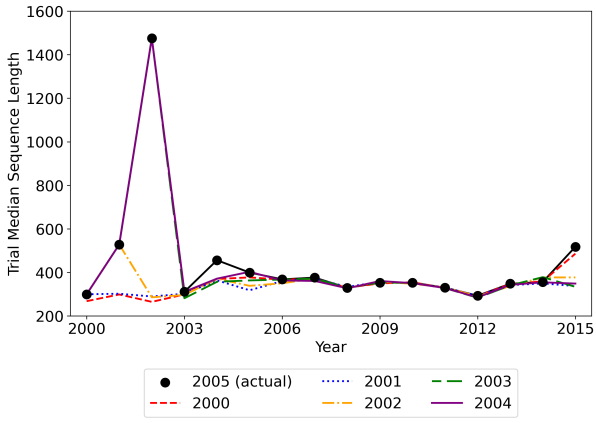


(b) JCSG, Explore-Then-Commit

(c) MCSG, UCB1+Time Discounting

(d) NESG, UCB1+Time Discounting

(e) NYSGRC*, UCB1+Time Discounting

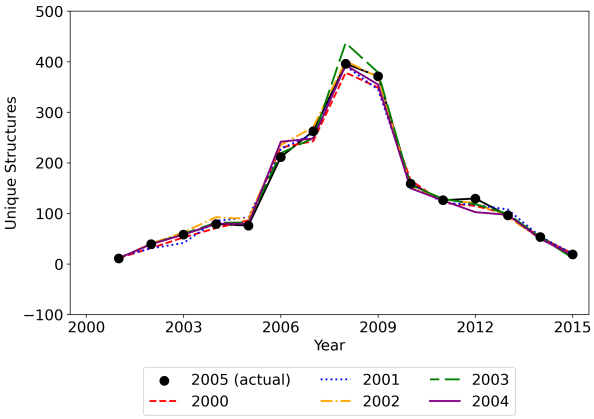


Note: See notes from Appendix Figure D8 for additional details.

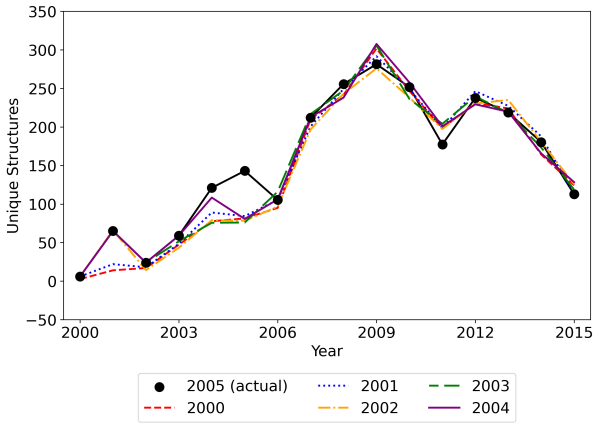
Figure D12 Number of Unique Structures Produced Under Alternative PSI Pilot Phase Duration

(a) JCSG, UCB1+Time Discounting

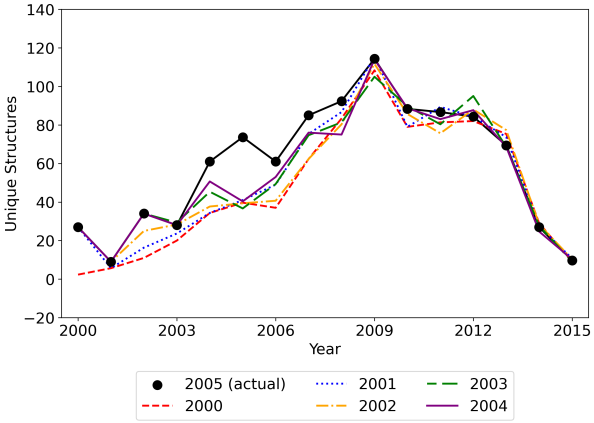
(b) JCSG, Explore-Then-Commit



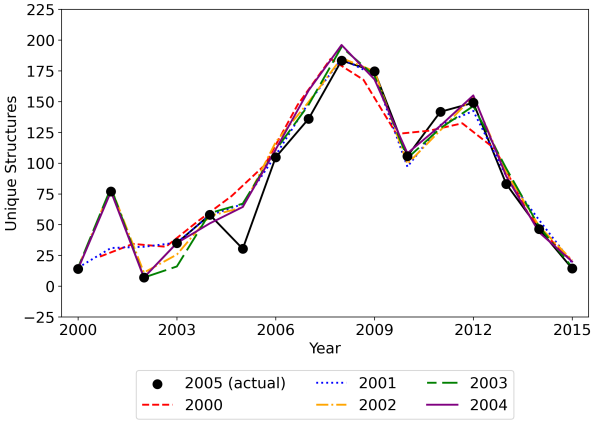
(c) MCSG, UCB1+Time Discounting



(d) NESG, UCB1+Time Discounting



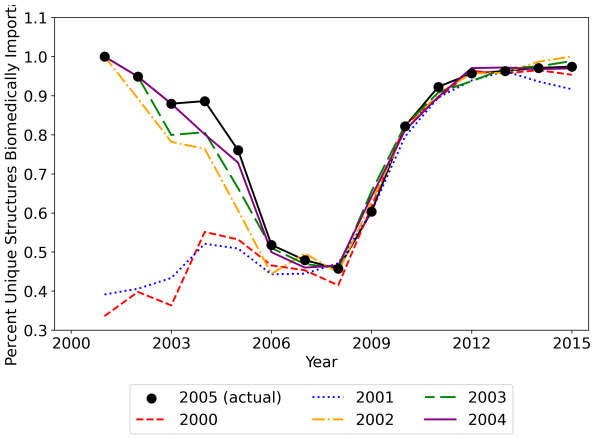
(e) NYSGRC*, UCB1+Time Discounting



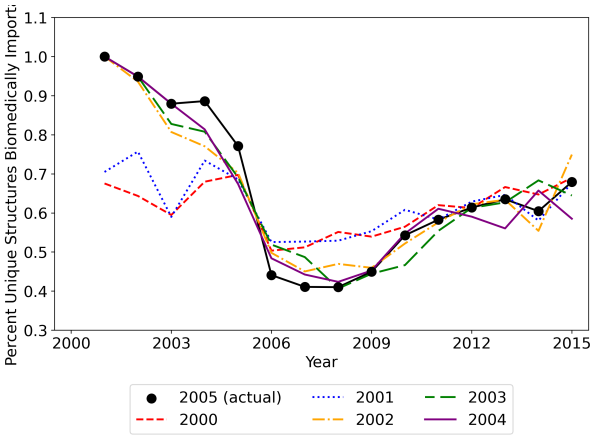
Note: See notes from Appendix Figure D8 for additional details.

Figure D13 Proportion of Biomedically Important Structures Under Alternative PSI Pilot Phase Duration

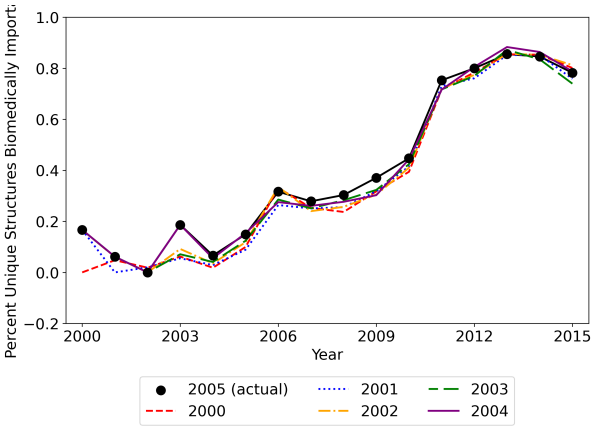
(a) JCSG, UCB1+Time Discounting



(b) JCSG, Explore-Then-Commit



(c) MCSG, UCB1+Time Discounting



(d) NESG, UCB1+Time Discounting

(e) NYSGRC*, UCB1+Time Discounting

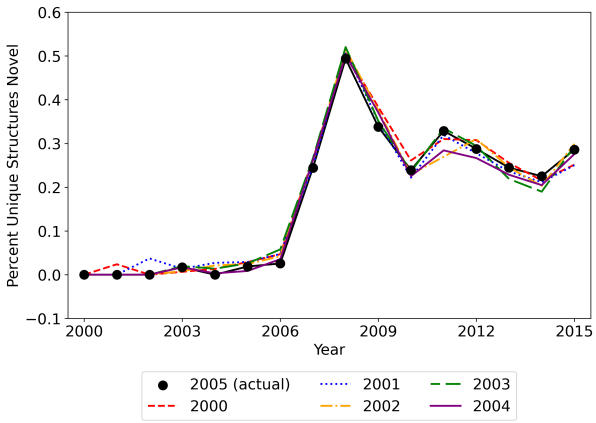
Note: See notes from Appendix Figure D8 for additional details.

Figure D14 Proportion of Novel Structures Under Alternative PSI Pilot Phase Duration

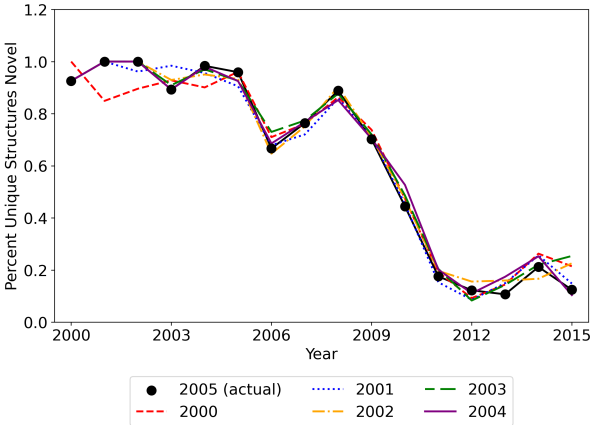
(a) JCSG, UCB1+Time Discounting

(b) JCSG, Explore-Then-Commit

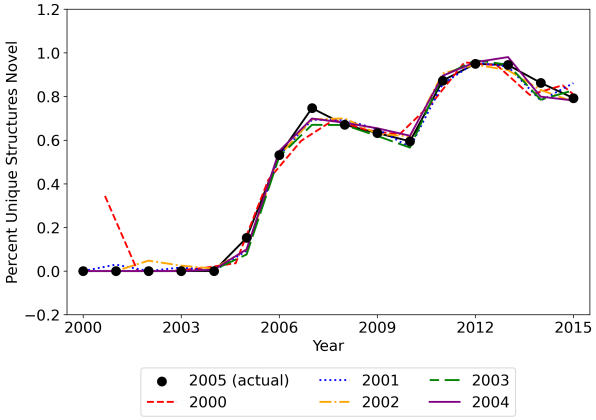
(c) MCSG, UCB1+Time Discounting



(d) NESG, UCB1+Time Discounting



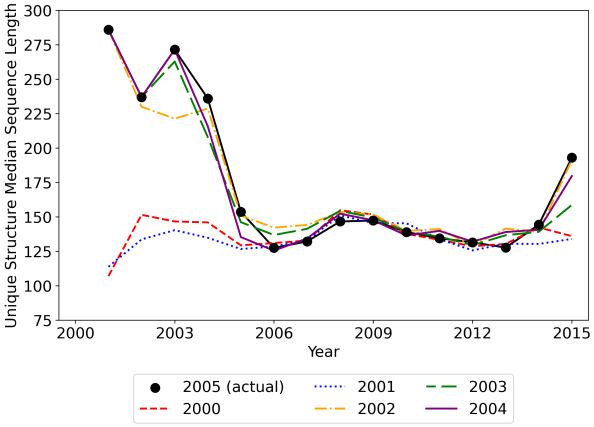
(e) NYSGRC*, UCB1+Time Discounting



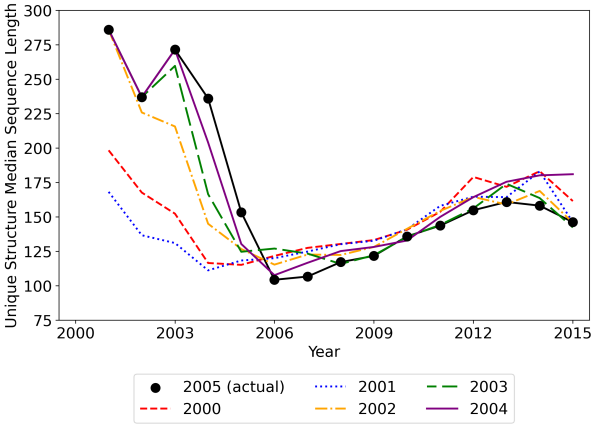
Note: See notes from Appendix Figure D8 for additional details.

Figure D15 Unique Structure Median Sequence Length Under Alternative PSI Pilot Phase Duration

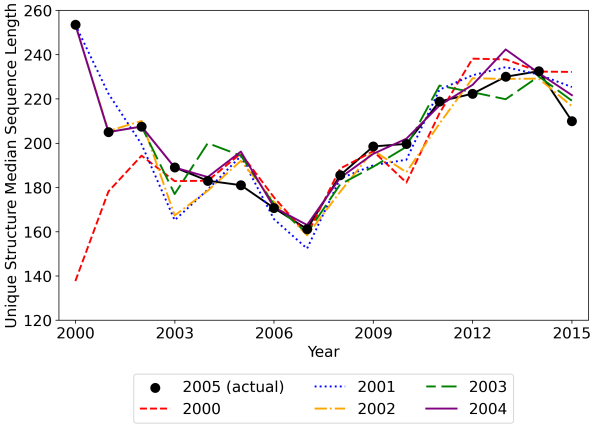
(a) JCSG, UCB1+Time Discounting



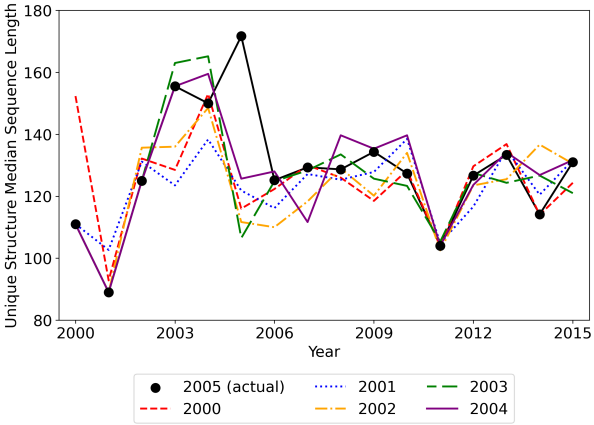
(b) JCSG, Explore-Then-Commit



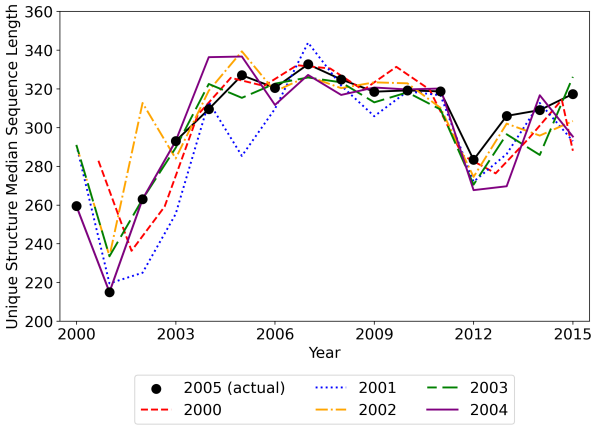
(c) MCSG, UCB1+Time Discounting



(d) NESG, UCB1+Time Discounting



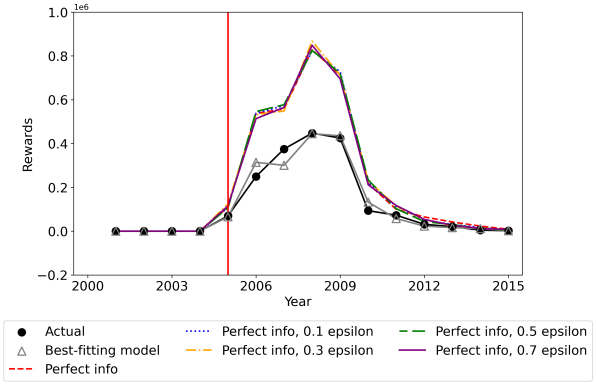
(e) NYSGRC*, UCB1+Time Discounting



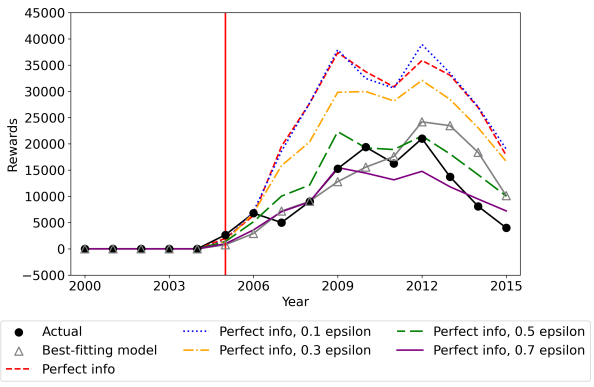
Note: See notes from Appendix Figure D8 for additional details.

Figure D16 Simulated Reward Trajectories Under Improved Information and Learning Model

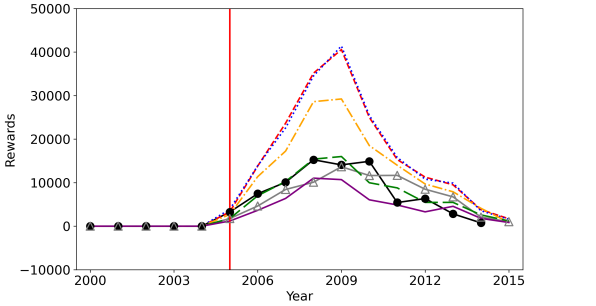
(a) JCSG, Greedy



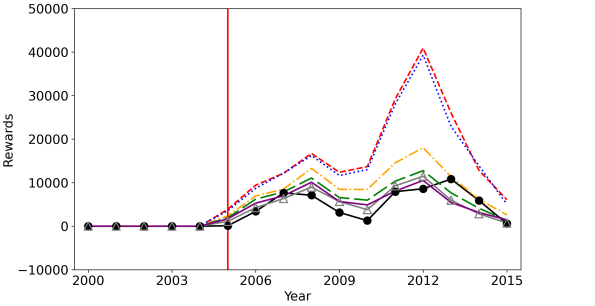
(b) MCSG, Greedy

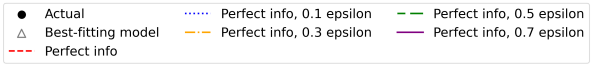


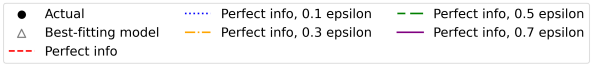
(c) NESG, Greedy



(d) NYSGRC*, Greedy



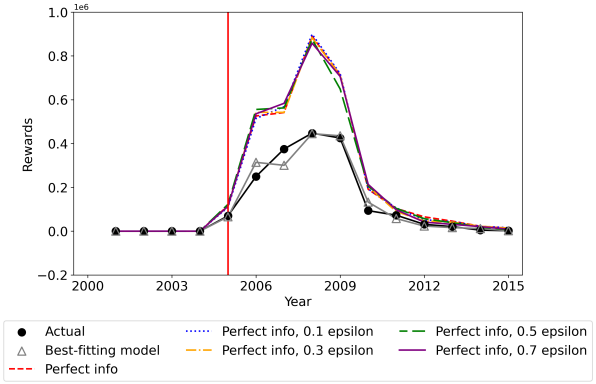




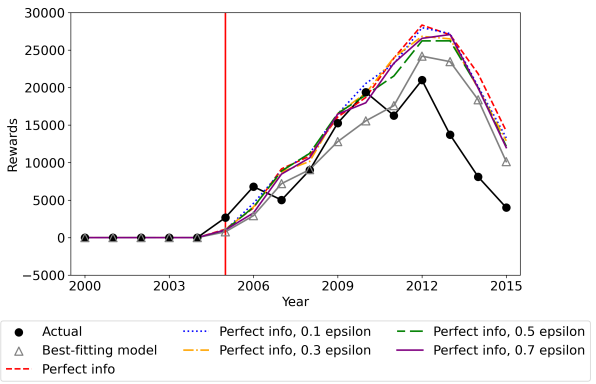
Note: The period 2000–2004 represents actual historical data (shown to the left of the red vertical line). Because the reward function is not estimated for this period, rewards are displayed as zero. The period from 2005 onward (to the right of the red vertical line) shows simulated rewards. Each line represents annual rewards under the allocation model indicated in the subfigure title, varying by information level and the magnitude of ϵ error, and averaged over three simulation runs.

Figure D17 Simulated Reward Trajectories Under Improved Information and Learning Model, Continued

(a) JCSG, Explore-Then-Commit

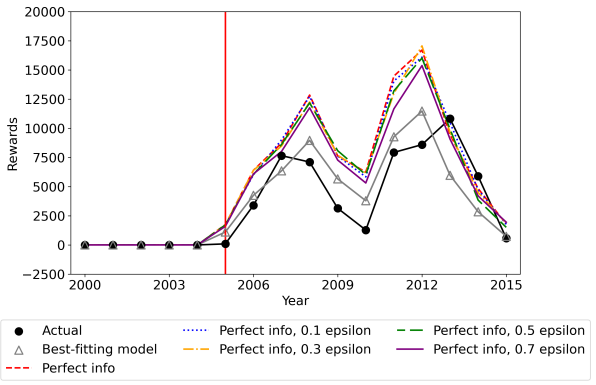


(b) MCSG, UCB1+Time Discounting



(c) NESG, UCB1+Time Discounting

(d) NYSGRC*, UCB1+Time Discounting



Note: See notes from Appendix Figure D16 for additional details.

Table D1: Allocation Model Estimates, JCSG

Model	Time period	θ_{quant}	θ_{novel}	$\theta_{prevStructZ}$	θ_{biomed}	$\theta_{prevPubZ}$	θ_{human}	$\theta_{eukaryote}$	$\theta_{membrane}$	λ_1	λ_2
Greedy	2005–2008	1059.52	1088.60	-164.21	-157.05	-353.40	-201.85	92.77	-100.87		
		[1056.14,1063.61]	[1083.90,1092.24]	[-165.28,-163.15]	[-157.78,-156.45]	[-354.98,-351.28]	[-202.91,-200.57]	[91.93,93.72]	[-101.04,-100.50]		
Greedy	2009–2015	128.77	-66.16	-13.05	205.46	3.01	105.10	16.53	87.02		
		[128.36,129.22]	[-66.38,-65.91]	[-13.14,-12.97]	[204.92,206.06]	[2.99,3.04]	[104.77,105.36]	[16.42,16.72]	[86.85,87.27]		
Gittins	2005–2008	293.32	392.45	33.91	-221.75	-246.10	-181.27	17.52	-18.69		
		[291.43,295.87]	[391.68,393.40]	[33.78,34.14]	[-222.92,-221.21]	[-247.11,-245.48]	[-182.76,-180.41]	[17.18,17.78]	[-19.24,-18.35]		
Gittins	2009–2015	95.79	-88.68	-10.80	154.94	-5.30	92.63	-0.50	85.00		
		[95.60,95.95]	[-88.77,-88.58]	[-10.84,-10.76]	[154.79,155.10]	[-5.32,-5.27]	[92.37,92.82]	[-0.59,-0.41]	[84.88,85.11]		
Thompson	2005–2008	3020.40	3218.08	162.60	1142.16	-660.54	328.89	399.25	404.27		
Sampling		[2977.52,3112.41]	[3133.93,3276.67]	[136.67,181.13]	[1126.54,1166.07]	[-674.28,-647.26]	[311.24,344.38]	[378.60,437.37]	[391.26,417.13]		
Thompson	2009–2015	805.69	116.56	101.36	858.99	28.82	609.82	763.01	154.51		
Sampling		[803.98,806.71]	[115.64,117.93]	[101.25,101.44]	[856.04,860.31]	[28.66,28.97]	[609.23,610.12]	[761.36,763.34]	[153.98,155.01]		
Explore-Then-	2005–2008	1282.54	1335.37	-223.41	-317.71	-466.76	-623.03	77.05	-123.54		
Commit		[1276.94,1287.53]	[1328.97,1341.83]	[-224.86,-222.00]	[-319.62,-316.53]	[-473.62,-462.00]	[-623.54,-622.02]	[75.63,78.48]	[-124.79,-121.95]		
Explore-Then-	2009–2015	212.37	-170.20	-8.26	336.31	9.29	202.93	-2.64	112.56		
Commit		[212.17,212.63]	[-170.45,-169.96]	[-8.39,-8.15]	[336.05,336.46]	[9.27,9.32]	[202.80,203.10]	[-2.75,-2.54]	[112.46,112.66]		
UCB1	2005–2008	1048.66	1072.68	-159.33	-174.05	-354.99	-203.13	89.41	-102.78	6.76	
		[1048.63,1048.71]	[1072.59,1072.78]	[-159.34,-159.30]	[-174.09,-174.00]	[-355.03,-354.96]	[-203.17,-203.07]	[89.36,89.46]	[-102.85,-102.75]	[6.75,6.76]	
UCB1	2009–2015	29.49	-127.33	-8.64	116.40	-5.33	60.86	-19.49	59.45	6.58	
		[29.42,29.54]	[-127.37,-127.29]	[-8.68,-8.63]	[116.39,116.43]	[-5.42,-5.25]	[60.83,60.94]	[-19.51,-19.48]	[59.40,59.60]	[6.57,6.58]	
1st-Degree	2005–2008	1059.70	1088.31	-166.04	-157.64	-353.44	-201.74	93.29	-100.96	-1.74e-02	
Polynomial		[1059.70,1059.71]	[1088.30,1088.32]	[-166.05,-166.03]	[-157.65,-157.63]	[-353.45,-353.44]	[-201.74,-201.73]	[93.29,93.30]	[-100.98,-100.96]	[-1.76e-02,-1.72e-02]	
1st-Degree	2009–2015	123.32	-66.03	-13.49	205.96	2.55	106.24	16.28	89.02	-2.35e-02	
Polynomial		[123.32,123.32]	[-66.03,-66.02]	[-13.49,-13.49]	[205.96,205.97]	[2.55,2.56]	[106.24,106.24]	[16.28,16.28]	[89.02,89.02]	[-2.36e-02,-2.35e-02]	
2nd-Degree	2005–2008	1059.20	1089.18	-165.58	-157.11	-352.35	-201.69	93.06	-101.13	-1.77e-02	5.21e-07
Polynomial		[1059.20,1059.20]	[1089.18,1089.18]	[-165.58,-165.58]	[-157.12,-157.11]	[-352.35,-352.35]	[-201.69,-201.69]	[93.06,93.06]	[-101.13,-101.13]	[-1.81e-02,-1.73e-02]	[5.05e-07,5.39e-07]
2nd-Degree	2009–2015	122.80	-66.32	-12.53	205.93	2.67	106.32	16.29	88.89	-2.33e-02	4.87e-07
Polynomial		[122.80,122.80]	[-66.33,-66.32]	[-12.53,-12.53]	[205.93,205.93]	[2.66,2.67]	[106.31,106.32]	[16.29,16.29]	[88.89,88.89]	[-2.34e-02,-2.32e-02]	[4.83e-07,4.91e-07]
Flexible	2005–2008	1060.48	1088.71	-164.19	-156.93	-353.52	-201.75	92.69	-101.04	-0.03	
Variance		[1060.47,1060.48]	[1088.70,1088.72]	[-164.20,-164.18]	[-156.94,-156.91]	[-353.53,-353.52]	[-201.75,-201.75]	[92.69,92.70]	[-101.05,-101.04]	[-0.03,-0.03]	
Flexible	2009–2015	49.78	-83.26	-16.18	118.08	-4.58	98.22	80.17	1.79		
Variance		[49.75,49.83]	[-83.32,-83.21]	[-16.21,-16.16]	[118.04,118.13]	[-4.59,-4.56]	[98.21,98.23]	[-48.76,-48.68]	[80.14,80.21]	[1.79,1.80]	
Flex Var+Time	2005–2008	1062.37	1090.21	-163.81	-157.71	-352.32	-206.84	94.64	-103.04	-0.09	11.94
Discounting		[1062.37,1062.38]	[1090.21,1090.21]	[-163.81,-163.81]	[-157.71,-157.71]	[-352.33,-352.32]	[-206.85,-206.84]	[94.64,94.65]	[-103.04,-103.04]	[-0.09,-0.09]	[11.94,11.94]
Flex Var+Time	2009–2015	47.34	76.03	-17.29	89.46	-11.47	76.65	-57.57	73.12	0.34	77.59
Discounting		[47.33,47.35]	[75.94,76.12]	[-17.51,-16.99]	[89.40,89.54]	[-11.49,-11.45]	[76.58,76.72]	[-57.59,-57.56]	[73.11,73.13]	[0.34,0.35]	[77.50,77.69]
UCB1+Time	2005–2008	1006.88	1013.19	-141.66	-203.35	-340.47	-207.69	85.18	-106.29	6.41	191.03
Discounting		[1006.84,1006.91]	[1013.09,1013.27]	[-141.73,-141.57]	[-203.39,-203.30]	[-340.59,-340.39]	[-207.76,-207.62]	[85.11,85.26]	[-106.38,-106.22]	[6.40,6.42]	[190.96,191.06]
UCB1+Time	2009–2015	8.47	-120.40	-8.96	89.11	-5.83	39.52	-35.49	60.03	6.86	76.46
Discounting		[8.34,8.62]	[-121.28,-120.11]	[-9.27,-8.74]	[88.88,89.41]	[-6.34,-5.31]	[39.16,39.87]	[-35.72,-35.18]	[59.58,60.33]	[6.84,6.88]	[76.07,76.65]

Note: 95% confidence intervals, calculated using the MCMC approach from Chernozhukov and Hong (2003), are provided in brackets. These intervals are nearly identical to those obtained using Procedure 1 from Chen et al. (2018). We do not use bootstrap methods for confidence interval estimation because the observations are not independent: since the lab's available choices on future dates depend on past decisions, resampling a day's observations would invalidate future observations generated under the observed history. The MCMC approach to likelihood estimation described in Chernozhukov and Hong (2003), Chen et al. (2018) is well-established and effective for such dependent-data settings, and is relatively straightforward to implement provided the likelihood function can be computed.

Table D2: Allocation Model Estimates, MCSG

Model	Time period	θ_{quant}	θ_{novel}	$\theta_{prevStructZ}$	θ_{biomed}	$\theta_{prevPubZ}$	θ_{human}	$\theta_{eukaryote}$	$\theta_{membrane}$	λ_1	λ_2
Greedy	2005–2008	164.70	106.25	-7.90	21.87	-12.48	15.29	15.94	37.82		
		[164.48,164.93]	[106.11,106.40]	[-7.94,-7.84]	[21.75,22.01]	[-12.82,-12.05]	[14.90,15.64]	[15.84,16.07]	[37.65,37.93]		
Greedy	2009–2015	206.88	70.40	-8.30	168.33	-4.75	-15.04	-17.15	41.42		
		[206.68,207.09]	[70.31,70.49]	[-8.44,-8.19]	[168.16,168.53]	[-4.83,-4.60]	[-15.19,-14.91]	[-17.26,-17.00]	[41.34,41.51]		
Gittins	2005–2008	90.33	38.05	-4.45	4.47	-7.11	13.27	12.02	15.61		
		[90.14,90.49]	[37.94,38.15]	[-4.47,-4.34]	[4.41,4.53]	[-7.30,-6.86]	[13.16,13.50]	[11.73,12.13]	[15.55,15.73]		
Gittins	2009–2015	104.21	22.35	-2.82	65.22	-1.96	-11.72	-16.65	13.65		
		[104.08,104.36]	[22.29,22.40]	[-2.86,-2.78]	[65.09,65.33]	[-2.03,-1.80]	[-11.95,-11.53]	[-16.69,-16.60]	[13.62,13.69]		
Thompson	2005–2008	318.16	-26.04	-22.72	30.53	-0.90	-26.34	11.03	1.59		
Sampling		[315.44,320.00]	[-26.69,-25.51]	[-23.26,-22.14]	[28.81,31.67]	[-1.69,-0.02]	[-28.62,-24.10]	[10.12,11.89]	[0.78,3.42]		
Thompson	2009–2015	432.94	41.04	-30.00	247.71	-2.94	11.04	4.83	61.75		
Sampling		[432.18,433.83]	[40.57,41.64]	[-30.33,-29.67]	[247.08,248.37]	[-3.22,-2.81]	[10.03,13.01]	[4.29,5.28]	[61.24,62.33]		
Explore-Then-	2005–2008	118.41	34.90	5.08	174.24	-16.83	23.34	9.99	47.43		
Commit		[117.93,118.82]	[34.74,35.06]	[4.81,5.41]	[173.65,174.89]	[-17.26,-16.38]	[22.99,23.83]	[9.81,10.20]	[47.21,47.69]		
Explore-Then-	2009–2015	178.61	83.97	0.19	223.86	1.93	-10.25	-44.39	33.95		
Commit		[178.36,178.86]	[83.66,84.23]	[0.01,0.34]	[223.43,224.34]	[1.09,2.36]	[-10.52,-10.00]	[-44.52,-44.26]	[33.81,34.18]		
UCB1	2005–2008	44.08	-3.82	2.73	18.75	2.78	13.46	-9.34	-9.10	6.18	
		[43.93,44.30]	[-4.05,-3.44]	[2.59,2.82]	[18.54,18.99]	[2.56,2.90]	[13.34,13.60]	[-9.42,-9.24]	[-9.21,-9.00]	[6.18,6.19]	
UCB1	2009–2015	67.36	26.88	13.26	111.92	14.16	-20.48	-65.14	14.82	5.30	
		[67.35,67.41]	[26.82,26.93]	[13.22,13.30]	[111.90,111.95]	[14.12,14.18]	[-20.51,-20.45]	[-65.27,-65.04]	[14.81,14.84]	[5.30,5.31]	
1st-Degree	2005–2008	76.38	104.23	-7.53	20.99	-8.39	16.24	11.77	35.03	-2.83e-02	
Polynomial		[76.35,76.42]	[104.23,104.23]	[-7.53,-7.53]	[20.99,21.00]	[-8.40,-8.39]	[16.24,16.25]	[11.77,11.78]	[35.01,35.04]	[-2.86e-02,-2.78e-02]	
1st-Degree	2009–2015	203.13	70.56	-8.89	167.59	-1.45	-14.27	-18.07	41.24	-1.38e-02	
Polynomial		[203.13,203.14]	[70.56,70.56]	[-8.89,-8.88]	[167.59,167.59]	[-1.45,-1.44]	[-14.27,-14.27]	[-18.07,-18.06]	[41.24,41.24]	[-1.40e-02,-1.37e-02]	
2nd-Degree	2005–2008	78.70	104.73	-5.19	23.72	-7.23	12.68	13.19	36.06	-2.74e-02	2.44e-06
Polynomial		[78.69,78.72]	[104.72,104.74]	[-5.21,-5.18]	[23.71,23.73]	[-7.25,-7.22]	[12.67,12.69]	[13.18,13.22]	[36.04,36.09]	[-2.81e-02,-2.70e-02]	[2.38e-06,2.49e-06]
2nd-Degree	2009–2015	201.82	70.74	-7.13	165.82	0.18	-13.27	-16.69	40.81	-1.35e-02	-2.40e-06
Polynomial		[201.82,201.82]	[70.74,70.75]	[-7.13,-7.13]	[165.82,165.82]	[0.17,0.18]	[-13.27,-13.27]	[-16.70,-16.69]	[40.81,40.82]	[-1.36e-02,-1.35e-02]	[-2.45e-06,-2.37e-06]
Flexible	2005–2008	149.09	98.55	-10.35	17.27	-11.44	12.74	11.01	37.74	0.23	
Variance		[149.09,149.11]	[98.54,98.57]	[-10.36,-10.35]	[17.25,17.28]	[-11.48,-11.39]	[12.73,12.75]	[11.00,11.03]	[37.71,37.76]	[0.23,0.23]	
Flexible	2009–2015	201.75	69.09	-7.97	166.21	-3.27	-14.91	-19.54	44.09	0.10	
Variance		[201.75,201.76]	[69.08,69.09]	[-7.97,-7.96]	[166.20,166.22]	[-3.28,-3.27]	[-14.92,-14.90]	[-19.55,-19.53]	[44.09,44.10]	[0.10,0.10]	
Flex Var+Time	2005–2008	145.51	94.77	-7.10	13.06	-8.50	11.20	12.04	37.37	0.01	13.70
Discounting		[145.50,145.53]	[94.75,94.80]	[-7.13,-7.08]	[13.04,13.08]	[-8.51,-8.49]	[11.18,11.22]	[12.03,12.06]	[37.35,37.38]	[0.01,0.02]	[13.66,13.71]
Flex Var+Time	2009–2015	201.14	70.79	-4.91	162.92	-0.95	-18.82	-17.81	41.05	-0.06	7.87
Discounting		[201.13,201.17]	[70.77,70.80]	[-4.93,-4.90]	[162.90,162.94]	[-0.96,-0.93]	[-18.83,-18.80]	[-17.83,-17.78]	[41.04,41.06]	[-0.07,-0.06]	[7.85,7.88]
UCB1+Time	2005–2008	44.26	-8.54	3.09	17.57	1.93	14.43	-6.52	-15.08	5.68	4.48
Discounting		[44.14,44.32]	[-8.81,-8.34]	[2.86,3.28]	[17.50,17.66]	[1.76,2.10]	[14.31,14.61]	[-6.81,-6.35]	[-15.15,-14.98]	[5.65,5.70]	[4.43,4.54]
UCB1+Time	2009–2015	49.49	23.19	15.06	89.49	8.75	-19.22	-65.52	12.93	4.87	3.82
Discounting		[49.46,49.53]	[23.15,23.22]	[15.01,15.11]	[89.46,89.52]	[8.73,8.78]	[-19.23,-19.19]	[-65.57,-65.49]	[12.86,12.98]	[4.86,4.87]	[3.81,3.83]

Note: The note from Appendix Table D1 applies.

Table D3: Allocation Model Estimates, NESG

Model	Time period	θ_{quant}	θ_{novel}	$\theta_{prevStructZ}$	θ_{biomed}	$\theta_{prevPubZ}$	θ_{human}	$\theta_{eukaryote}$	$\theta_{membrane}$	λ_1	λ_2
Greedy	2005–2008	271.96	206.48	-1.29	50.51	-15.51	65.81	86.17	97.42		
		[270.19,273.51]	[205.92,207.06]	[-1.48,-1.09]	[50.05,50.98]	[-15.82,-15.33]	[65.22,66.41]	[85.77,86.58]	[97.02,97.96]		
Greedy	2009–2015	228.76	33.74	43.99	138.23	2.71	177.79	64.53	68.58		
		[228.56,229.03]	[33.59,33.95]	[43.90,44.14]	[138.09,138.47]	[2.66,2.75]	[177.60,178.02]	[64.41,64.66]	[68.47,68.70]		
Gittins	2005–2008	151.65	77.68	1.70	15.87	-5.60	30.25	8.88	33.56		
		[151.07,152.17]	[77.37,78.02]	[1.65,1.77]	[15.52,16.20]	[-5.67,-5.54]	[30.13,30.35]	[8.81,8.97]	[33.46,33.69]		
Gittins	2009–2015	154.27	-7.36	-0.27	35.93	-3.10	70.51	-17.92	20.34		
		[153.98,154.52]	[-7.47,-7.29]	[-0.32,-0.21]	[35.78,36.06]	[-3.12,-3.06]	[70.35,70.71]	[-18.05,-17.80]	[20.28,20.40]		
Thompson	2005–2008	376.31	138.40	42.00	168.27	-10.44	102.72	40.88	24.04		
Sampling		[372.79,380.59]	[136.18,141.59]	[40.23,43.64]	[165.83,171.87]	[-11.06,-9.77]	[100.20,106.97]	[38.25,43.19]	[20.35,25.98]		
Thompson	2009–2015	609.20	-138.37	27.63	19.97	6.14	114.90	160.26	50.92		
Sampling		[606.26,612.66]	[-141.15,-135.95]	[27.00,28.04]	[17.88,22.14]	[5.77,6.43]	[113.07,116.59]	[158.74,161.40]	[49.49,52.74]		
Explore-Then-	2005–2008	212.69	152.77	1.72	42.21	-13.86	32.79	43.43	81.99		
Commit		[211.36,214.10]	[151.48,153.88]	[0.98,2.40]	[41.37,43.01]	[-14.26,-13.52]	[32.34,33.29]	[43.01,43.89]	[81.44,82.68]		
Explore-Then-	2009–2015	250.31	11.81	35.89	139.04	-2.68	199.29	59.29	59.88		
Commit		[250.05,250.61]	[11.62,12.19]	[35.75,36.00]	[138.56,139.86]	[-2.72,-2.62]	[199.05,199.51]	[58.84,59.65]	[59.61,60.14]		
UCB1	2005–2008	99.59	18.93	24.12	30.85	-5.22	33.16	-24.52	40.97	5.37	
		[99.54,99.64]	[18.66,19.11]	[23.87,24.38]	[30.06,31.45]	[-5.48,-4.90]	[33.07,33.22]	[-24.89,-24.24]	[40.79,41.26]	[5.37,5.39]	
UCB1	2009–2015	82.96	-16.60	40.08	94.54	-7.63	125.52	-6.12	43.09	5.25	
		[82.79,83.14]	[-16.92,-16.34]	[40.04,40.11]	[94.50,94.58]	[-7.65,-7.61]	[125.42,125.66]	[-6.21,-6.00]	[42.80,43.35]	[5.25,5.26]	
1st-Degree	2005–2008	236.12	183.55	0.67	43.03	-12.35	63.49	73.16	95.52	-4.97e-01	
Polynomial		[236.11,236.16]	[183.54,183.55]	[0.64,0.68]	[42.97,43.06]	[-12.38,-12.32]	[63.46,63.53]	[73.14,73.17]	[95.48,95.57]	[-4.99e-01,-4.96e-01]	
1st-Degree	2009–2015	198.16	-49.86	38.90	131.86	1.98	169.26	53.93	65.55	-2.96e-01	
Polynomial		[198.15,198.16]	[-49.88,-49.82]	[38.89,38.90]	[131.86,131.86]	[1.98,1.99]	[169.25,169.26]	[53.93,53.93]	[65.54,65.57]	[-2.96e-01,-2.95e-01]	
2nd-Degree	2005–2008	221.14	173.40	7.03	37.46	-11.10	58.69	67.01	92.44	-4.65e-01	-4.16e-05
Polynomial		[221.11,221.20]	[173.39,173.41]	[7.02,7.04]	[37.43,37.49]	[-11.12,-11.07]	[58.65,58.70]	[66.99,67.03]	[92.43,92.45]	[-4.66e-01,-4.65e-01]	[-4.18e-05,-4.15e-05]
2nd-Degree	2009–2015	179.45	-58.89	36.64	117.10	0.96	101.37	-78.08	22.57	-2.69e-01	4.58e-05
Polynomial		[179.44,179.48]	[-58.91,-58.86]	[36.57,36.69]	[117.05,117.19]	[0.89,1.04]	[101.06,101.59]	[-78.28,-77.85]	[22.29,22.76]	[-2.70e-01,-2.69e-01]	[4.52e-05,4.63e-05]
Flexible	2005–2008	267.63	204.67	0.33	50.38	-12.91	64.54	82.78	99.86	0.13	
Variance		[267.61,267.67]	[204.65,204.71]	[0.32,0.34]	[50.38,50.40]	[-12.92,-12.89]	[64.52,64.55]	[82.77,82.80]	[99.85,99.89]	[0.13,0.14]	
Flexible	2009–2015	224.05	33.20	44.08	136.46	2.03	178.09	63.58	68.88	0.09	
Variance		[224.05,224.06]	[33.20,33.22]	[44.07,44.08]	[136.46,136.47]	[2.02,2.04]	[178.08,178.11]	[63.57,63.59]	[68.88,68.89]	[0.09,0.09]	
Flex Var+Time	2005–2008	267.25	203.61	3.34	47.61	-10.47	60.59	83.88	98.71	0.02	10.99
Discounting		[267.25,267.26]	[203.61,203.62]	[3.33,3.35]	[47.61,47.62]	[-10.48,-10.46]	[60.58,60.60]	[83.87,83.89]	[98.70,98.72]	[0.02,0.02]	[10.98,10.99]
Flex Var+Time	2009–2015	223.89	31.38	45.05	134.71	2.35	173.11	63.45	69.01	-0.06	10.95
Discounting		[223.88,223.89]	[31.35,31.39]	[45.04,45.05]	[134.70,134.72]	[2.34,2.37]	[173.10,173.11]	[63.44,63.45]	[69.00,69.01]	[-0.06,-0.06]	[10.94,10.96]
UCB1+Time	2005–2008	99.05	18.18	22.10	32.55	-4.70	33.62	-24.69	39.11	5.11	2.39
Discounting		[98.94,99.18]	[18.09,18.25]	[21.96,22.25]	[32.31,32.65]	[-4.78,-4.63]	[33.52,33.77]	[-24.78,-24.57]	[38.94,39.23]	[5.10,5.12]	[2.37,2.41]
UCB1+Time	2009–2015	61.07	-2.61	0.44	59.70	0.72	40.57	-16.71	38.09	4.83	4.65
Discounting		[60.90,61.40]	[-2.77,-2.44]	[0.22,0.65]	[59.42,59.91]	[0.62,0.83]	[40.12,41.08]	[-17.23,-16.07]	[37.89,38.24]	[4.82,4.84]	[4.61,4.67]

Note: The note from Appendix Table D1 applies.

Table D4: Allocation Model Estimates, NYSGRC*

Model	Time period	θ_{quant}	θ_{novel}	$\theta_{prevStructZ}$	θ_{biomed}	$\theta_{prevPubZ}$	θ_{human}	$\theta_{eukaryote}$	$\theta_{membrane}$	λ_1	λ_2
Greedy	2005–2008	70.48	55.31	-13.02	30.29	2.11	4.72	20.65	12.52		
		[70.21,70.75]	[55.13,55.48]	[-13.13,-12.92]	[30.19,30.39]	[1.99,2.20]	[4.60,4.83]	[20.53,20.78]	[12.42,12.61]		
Greedy	2009–2015	-9.85	171.53	8.51	155.18	1.90	-22.83	2.46	61.47		
		[-9.95,-9.80]	[171.37,171.73]	[8.49,8.55]	[155.04,155.35]	[1.88,1.92]	[-22.91,-22.75]	[2.41,2.54]	[61.36,61.59]		
Gittins	2005–2008	24.96	34.67	-7.99	30.35	3.78	10.33	8.66	10.55		
		[24.87,25.08]	[34.60,34.79]	[-8.06,-7.93]	[30.28,30.48]	[3.74,3.82]	[10.24,10.41]	[8.58,8.78]	[10.47,10.62]		
Gittins	2009–2015	-1.70	64.22	1.89	62.52	0.72	-5.86	16.07	28.83		
		[-1.75,-1.66]	[64.14,64.29]	[1.86,1.91]	[62.45,62.66]	[0.66,0.77]	[-5.91,-5.79]	[16.03,16.11]	[28.79,28.92]		
Thompson	2005–2008	69.26	93.59	-2.86	79.12	13.43	16.61	11.43	25.83		
Sampling		[68.72,69.91]	[92.96,94.33]	[-3.42,-2.44]	[78.38,79.86]	[13.01,14.33]	[15.69,17.74]	[10.62,12.07]	[25.21,26.27]		
Thompson	2009–2015	47.00	133.61	0.95	194.60	30.20	22.32	80.15	100.35		
Sampling		[46.36,47.49]	[132.73,134.38]	[0.76,1.10]	[193.62,195.34]	[29.73,30.53]	[21.28,23.04]	[79.57,80.89]	[99.68,101.01]		
Explore-Then-	2005–2008	67.66	52.71	-18.63	30.24	4.16	0.86	16.43	20.21		
Commit		[67.46,67.86]	[52.40,52.93]	[-18.85,-18.31]	[30.06,30.46]	[4.07,4.32]	[0.58,1.08]	[16.13,16.62]	[20.03,20.37]		
Explore-Then-	2009–2015	-71.24	169.91	7.80	168.15	0.38	20.14	34.14	86.73		
Commit		[-71.49,-70.77]	[169.35,170.17]	[7.76,7.87]	[167.80,168.40]	[0.29,0.48]	[19.81,20.40]	[33.90,34.41]	[86.52,86.94]		
UCB1	2005–2008	14.61	38.67	4.19	32.93	13.95	2.45	-9.21	11.08	4.54	
		[14.56,14.69]	[38.52,38.85]	[4.11,4.29]	[32.89,32.99]	[13.83,14.01]	[2.37,2.54]	[-9.24,-9.18]	[11.05,11.13]	[4.53,4.54]	
UCB1	2009–2015	-77.11	81.81	9.48	88.05	1.98	-8.23	-21.78	61.30	5.39	
		[-77.12,-77.09]	[81.67,81.97]	[9.40,9.55]	[87.93,88.15]	[1.97,2.00]	[-8.34,-8.11]	[-21.80,-21.77]	[61.02,61.49]	[5.39,5.40]	
1st-Degree	2005–2008	70.54	55.32	-13.17	30.38	2.16	4.73	20.65	12.55	1.94e-03	
Polynomial		[70.54,70.54]	[55.32,55.32]	[-13.17,-13.16]	[30.38,30.38]	[2.16,2.16]	[4.73,4.73]	[20.65,20.65]	[12.55,12.56]	[1.93e-03,1.96e-03]	
1st-Degree	2009–2015	-13.99	170.40	7.12	155.84	3.72	-23.34	2.26	60.86	-1.24e-02	
Polynomial		[-14.00,-13.99]	[170.40,170.41]	[7.12,7.13]	[155.84,155.84]	[3.72,3.73]	[-23.34,-23.34]	[2.25,2.26]	[60.85,60.86]	[-1.26e-02,-1.22e-02]	
2nd-Degree	2005–2008	70.12	58.95	-12.95	28.12	1.94	4.68	23.46	13.89	-4.96e-03	1.83e-06
Polynomial		[70.11,70.12]	[58.95,58.96]	[-12.95,-12.95]	[28.12,28.13]	[1.93,1.94]	[4.68,4.68]	[23.46,23.47]	[13.89,13.89]	[-5.01e-03,-4.92e-03]	[1.81e-06,1.85e-06]
2nd-Degree	2009–2015	-14.05	170.74	7.15	155.69	3.43	-24.22	1.49	61.04	-1.42e-02	2.00e-06
Polynomial		[-14.05,-14.04]	[170.74,170.75]	[7.15,7.15]	[155.69,155.70]	[3.43,3.43]	[-24.22,-24.21]	[1.49,1.50]	[61.04,61.04]	[-1.43e-02,-1.40e-02]	[1.98e-06,2.02e-06]
Flexible	2005–2008	60.41	52.20	-14.73	29.53	3.77	5.52	16.86	16.56	0.12	
Variance		[60.39,60.42]	[52.19,52.22]	[-14.74,-14.72]	[29.51,29.54]	[3.76,3.78]	[5.50,5.57]	[16.85,16.88]	[16.55,16.57]	[0.11,0.12]	
Flexible	2009–2015	-18.29	167.39	4.47	149.72	3.56	-22.20	0.18	63.81	0.12	
Variance		[-18.29,-18.28]	[167.38,167.39]	[4.46,4.48]	[149.71,149.73]	[3.55,3.57]	[-22.21,-22.20]	[0.18,0.18]	[63.81,63.82]	[0.12,0.12]	
Flex Var+Time	2005–2008	13.33	17.67	-3.99	17.24	4.67	4.97	2.81	6.22	1.30	4.60
Discounting		[13.23,13.41]	[17.62,17.75]	[-4.02,-3.96]	[17.23,17.25]	[4.65,4.69]	[4.91,5.01]	[2.74,2.85]	[6.16,6.31]	[1.29,1.31]	[4.56,4.62]
Flex Var+Time	2009–2015	-21.41	166.28	6.69	145.19	3.86	-26.94	-0.02	62.79	0.01	10.26
Discounting		[-21.41,-21.40]	[166.28,166.29]	[6.68,6.70]	[145.18,145.22]	[3.85,3.86]	[-26.95,-26.93]	[-0.03,-0.01]	[62.77,62.80]	[0.01,0.01]	[10.26,10.27]
UCB1+Time	2005–2008	15.47	37.83	6.07	35.55	20.72	4.10	-11.31	11.33	4.16	2.87
Discounting		[15.39,15.58]	[37.75,37.87]	[5.89,6.21]	[35.50,35.62]	[20.66,20.80]	[3.98,4.22]	[-11.40,-11.19]	[11.15,11.56]	[4.15,4.16]	[2.85,2.88]
UCB1+Time	2009–2015	-76.48	76.51	6.89	83.10	1.48	-6.51	-20.70	59.14	4.87	4.14
Discounting		[-76.53,-76.45]	[76.44,76.61]	[6.81,6.97]	[83.05,83.13]	[1.43,1.54]	[-6.55,-6.47]	[-20.76,-20.65]	[59.10,59.17]	[4.87,4.87]	[4.13,4.15]

Note: The note from Appendix Table D1 applies. *Results for NYSGRC should be interpreted with caution due to data quality concerns.

Table D5: Simulated Allocations and Outcomes Under Alternative Allocation Models, JCSG

Model	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
Greedy	11,531	32.8	50.2	2,075	0.55	0.48	1.00	1.00	168.7	128.7	2,245,884.0
Gittins	11,764	32.2	49.8	1,831	0.55	0.49	1.00	0.99	208.0	140.0	1,976,002.0
Thompson Sampling	12,134	31.2	48.2	2,093	0.55	0.49	1.00	1.00	182.0	130.0	2,186,105.8
Explore-Then-Commit	40,881	9.3	26.0	2,086	0.59	0.53	0.99	0.98	176.7	129.3	2,313,911.1
Best-fitting model	40,881	9.3	19.6	1,837	0.67	0.74	0.92	0.86	211.3	144.0	1,803,448.2
Actual	40,881	9.3	28.4	1,509	0.71	0.67	0.92	0.95	242.0	219.0	1,791,493.8

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D1. All other notes from Table 6 also apply.

Table D6: Simulated Allocations and Outcomes Under Alternative Allocation Models, MCSG

Model	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
Greedy	14,652	10.8	57.9	1,372	0.09	0.19	0.04	0.05	225.7	183.3	40,729.1
Gittins	16,467	9.6	47.1	1,825	0.22	0.31	0.05	0.05	200.7	173.0	81,840.6
Thompson Sampling	23,375	6.8	34.2	2,127	0.22	0.29	0.13	0.11	188.7	166.3	101,402.2
Explore-Then-Commit	77,539	2.0	18.2	1,933	0.28	0.32	0.34	0.25	225.7	190.7	80,854.7
Best-fitting model	77,483	2.0	8.0	2,481	0.50	0.48	0.33	0.24	243.0	197.3	142,065.2
Actual	77,200	2.1	8.6	2,203	0.57	0.54	0.29	0.21	277.0	220.0	121,219.6

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D2. All other notes from Table 6 also apply.

Table D7: Simulated Allocations and Outcomes Under Alternative Allocation Models, NESG

Model	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
Greedy	12,343	10.8	51.1	595	0.08	0.21	0.96	0.91	188.0	137.2	28,394.8
Gittins	13,753	9.7	35.1	756	0.16	0.25	0.93	0.89	187.3	132.7	50,679.0
Thompson Sampling	18,166	7.3	24.9	806	0.19	0.30	0.89	0.84	188.7	134.7	59,726.7
Explore-Then-Commit	59,953	2.2	16.9	736	0.22	0.33	0.85	0.79	236.0	141.3	43,663.4
Best-fitting model	59,953	2.2	4.6	980	0.37	0.50	0.69	0.61	207.7	127.0	80,181.6
Actual	59,946	2.2	3.2	1,063	0.32	0.32	0.76	0.81	209.0	134.0	80,384.7

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D3. All other notes from Table 6 also apply.

Table D8: Simulated Allocations and Outcomes Under Alternative Allocation Models, NYSGRC*

Model	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
Greedy	5,192	28.7	83.1	772	0.18	0.15	0.40	0.24	370.0	294.2	-1,932.0
Gittins	5,981	25.0	75.2	867	0.21	0.20	0.43	0.32	357.7	290.2	3,853.5
Thompson Sampling	17,727	8.4	37.0	1,035	0.50	0.32	0.67	0.52	359.3	311.5	32,895.6
Explore-Then-Commit	59,734	2.5	18.3	1,065	0.59	0.39	0.72	0.55	352.0	310.2	37,834.0
Best-fitting model	59,733	2.5	9.9	1,363	0.71	0.48	0.78	0.63	336.3	308.7	60,405.2
Actual	59,734	2.5	24.6	1,334	0.75	0.49	0.76	0.66	361.0	319.0	56,485.5

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D4. All other notes from Table 6 also apply.

Table D9: Simulated Allocations and Outcomes (2005–2015) Under Alternative Allocation Models with Shortened Pilot Phase, JCSG

Pilot Phase ends at the start of	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
UCB1+Time Discounting											
2000	40,881	9.3	15.1	2,013	0.60	0.66	0.92	0.87	213.0	139.3	1,946,582.7
2001	40,881	9.3	15.2	2,032	0.60	0.65	0.92	0.87	209.3	136.3	1,933,399.8
2002	40,881	9.3	17.6	1,919	0.65	0.71	0.92	0.86	220.3	148.3	1,894,864.0
2003	40,881	9.3	17.7	1,848	0.67	0.72	0.92	0.87	225.0	145.3	1,877,904.2
2004	40,881	9.3	18.8	1,887	0.67	0.73	0.92	0.86	215.7	143.3	1,860,366.9
2005 (actual)	40,881	9.3	19.6	1,837	0.67	0.74	0.92	0.86	211.3	144.0	1,803,448.2
Explore-Then-Commit											
2000	40,881	9.3	24.0	2,024	0.59	0.57	0.99	0.98	177.3	135.0	2,519,106.1
2001	40,881	9.3	24.1	2,063	0.59	0.58	0.99	0.98	175.0	133.3	2,587,281.4
2002	40,881	9.3	24.9	2,133	0.60	0.54	0.99	0.99	184.3	132.7	2,413,061.9
2003	40,881	9.3	25.1	2,123	0.59	0.53	0.99	0.99	184.7	131.3	2,319,365.9
2004	40,881	9.3	25.5	2,045	0.60	0.53	0.99	0.98	184.3	131.3	2,277,073.7
2005 (actual)	40,881	9.3	26.0	2,086	0.59	0.53	0.99	0.98	176.7	129.3	2,313,911.1

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D1. All other notes from Table 7 also apply.

Table D10: Simulated Allocations and Outcomes (2005–2015) Under Shortened Pilot Phase, MCSG

Pilot Phase ends at the start of	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
UCB1+Time Discounting											
2000	77,490	2.0	8.1	2,307	0.50	0.49	0.34	0.27	243.7	198.7	140,076.8
2001	77,484	2.0	7.9	2,366	0.50	0.48	0.33	0.25	243.7	196.7	144,311.5
2002	77,483	2.0	8.0	2,336	0.50	0.49	0.33	0.26	244.7	195.7	141,106.2
2003	77,493	2.0	7.9	2,404	0.50	0.48	0.33	0.26	244.0	196.0	142,352.0
2004	77,486	2.0	7.8	2,439	0.50	0.48	0.34	0.24	244.3	200.0	144,104.9
2005 (actual)	77,483	2.0	8.0	2,481	0.50	0.48	0.33	0.24	243.0	197.3	142,065.2

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D2. All other notes from Table 7 also apply.

Table D11: Simulated Allocations and Outcomes (2005–2015) Under Shortened Pilot Phase, NESG

Pilot Phase ends at the start of	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
UCB1+Time Discounting											
2000	59,953	2.2	4.9	789	0.37	0.54	0.70	0.55	201.3	120.0	75,259.1
2001	59,953	2.2	4.7	868	0.37	0.55	0.70	0.56	199.7	121.0	81,617.1
2002	59,953	2.2	5.0	856	0.37	0.53	0.70	0.58	202.3	120.3	78,461.7
2003	59,953	2.2	4.7	891	0.37	0.52	0.70	0.59	199.0	122.7	78,903.6
2004	59,953	2.2	4.9	899	0.37	0.51	0.70	0.59	205.0	125.3	76,890.1
2005 (actual)	59,953	2.2	4.6	980	0.37	0.50	0.69	0.61	207.7	127.0	80,181.6

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D3. All other notes from Table 7 also apply.

Table D12: Simulated Allocations and Outcomes (2005–2015) Under Shortened Pilot Phase, NYSGRC*

Pilot Phase ends at the start of	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
UCB1+Time Discounting											
2000	59,733	2.5	7.0	1,406	0.69	0.46	0.78	0.61	333.3	310.2	60,236.2
2001	59,734	2.5	7.3	1,385	0.68	0.47	0.78	0.62	331.7	306.7	60,186.7
2002	59,733	2.5	7.3	1,424	0.69	0.45	0.78	0.62	331.0	306.5	60,407.6
2003	59,733	2.5	8.3	1,409	0.70	0.46	0.78	0.61	333.0	306.2	59,073.5
2004	59,733	2.5	8.3	1,432	0.70	0.46	0.78	0.62	333.7	305.7	61,716.4
2005 (actual)	59,733	2.5	9.9	1,363	0.71	0.48	0.78	0.63	336.3	308.7	60,405.2

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D4. All other notes from Table 7 also apply.

Table D13: Simulated Allocations and Outcomes (2005–2015) Under Improved Information and Learning Models, JCSG

	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
Greedy											
Perfect information	11,375	33.3	40.1	2,921	0.62	0.59	0.91	0.89	135.7	122.0	3,201,510.3
Perfect information, 0.1ϵ	11,712	32.3	44.4	2,790	0.59	0.56	0.94	0.91	140.3	120.7	3,208,911.7
Perfect information, 0.3ϵ	11,600	32.6	50.1	2,635	0.55	0.51	0.98	0.96	147.0	120.3	3,226,625.0
Perfect information, 0.5ϵ	11,503	32.9	51.3	2,596	0.55	0.50	0.99	0.98	148.3	121.0	3,202,628.4
Perfect information, 0.7ϵ	11,487	32.9	51.7	2,569	0.54	0.50	0.99	0.98	148.0	123.0	3,167,113.1
Explore-Then-Commit											
Perfect information	40,881	9.3	23.6	2,822	0.64	0.59	0.91	0.90	142.3	124.2	3,228,878.0
Perfect information, 0.1ϵ	40,881	9.3	25.2	2,732	0.61	0.57	0.93	0.92	147.0	124.7	3,241,963.2
Perfect information, 0.3ϵ	40,881	9.3	27.9	2,635	0.58	0.53	0.98	0.96	153.7	125.3	3,209,321.0
Perfect information, 0.5ϵ	40,881	9.3	28.3	2,595	0.57	0.53	0.98	0.97	154.0	126.0	3,195,262.6
Perfect information, 0.7ϵ	40,881	9.3	28.4	2,602	0.57	0.51	0.99	0.98	154.0	125.3	3,209,595.7
Best-fitting model	40,881	9.3	19.6	1,837	0.67	0.74	0.92	0.86	211.3	144.0	1,803,448.2
Actual	40,881	9.3	28.4	1,509	0.71	0.67	0.92	0.95	242.0	219.0	1,791,493.8

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D1. All other notes from Table 8 also apply.

Table D14: Simulated Allocations and Outcomes (2005–2015) Under Improved Information and Learning Models, MCSG

	Projects attempted	Trials per project		Unique structures	% Biomedical		% Novel		Median sequence length		Total Rewards
		Avg	Std		trials	structures	trials	structures	trials	structures	
Greedy											
Perfect information	21,735	7.3	15.5	4,175	0.65	0.54	0.24	0.22	198.0	183.3	271,166.2
Perfect information, 0.1ϵ	21,977	7.2	15.3	4,195	0.65	0.55	0.24	0.22	198.0	183.0	274,479.3
Perfect information, 0.3ϵ	20,988	7.6	25.4	3,576	0.54	0.53	0.26	0.22	202.0	183.0	232,668.9
Perfect information, 0.5ϵ	18,682	8.5	41.5	2,608	0.37	0.49	0.27	0.21	222.3	187.7	152,949.5
Perfect information, 0.7ϵ	17,288	9.2	49.3	2,061	0.28	0.43	0.28	0.20	220.7	190.7	106,941.5
UCB1+Time Discounting											
Perfect information	77,485	2.0	6.3	2,874	0.52	0.52	0.35	0.25	248.0	201.0	175,214.6
Perfect information, 0.1ϵ	77,481	2.0	6.3	2,863	0.52	0.51	0.35	0.25	248.0	200.7	174,495.5
Perfect information, 0.3ϵ	77,483	2.0	6.3	2,842	0.51	0.50	0.35	0.26	248.7	200.0	169,960.5
Perfect information, 0.5ϵ	77,482	2.0	6.5	2,816	0.51	0.50	0.35	0.26	249.0	202.3	166,883.1
Perfect information, 0.7ϵ	77,476	2.0	6.9	2,803	0.51	0.50	0.35	0.26	250.0	199.0	166,670.7
Best-fitting model	77,483	2.0	8.0	2,481	0.50	0.48	0.33	0.24	243.0	197.3	142,065.2
Actual	77,200	2.1	8.6	2,203	0.57	0.54	0.29	0.21	277.0	220.0	121,219.6

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D2. All other notes from Table 8 also apply.

Table D15: Simulated Allocations and Outcomes (2005–2015) Under Improved Information and Learning Models, NESG

	Projects	Trials per project		Unique	% Biomedical		% Novel		Median sequence length		Total
	attempted	Avg	Std	structures	trials	structures	trials	structures	trials	structures	Rewards
Greedy											
Perfect information	14,771	9.0	19.5	1,765	0.59	0.55	0.59	0.61	125.7	123.0	183,046.6
Perfect information, 0.1ϵ	14,818	9.0	19.2	1,759	0.59	0.55	0.60	0.62	126.0	123.8	183,260.7
Perfect information, 0.3ϵ	14,631	9.1	28.7	1,502	0.45	0.53	0.69	0.64	144.7	124.7	144,193.2
Perfect information, 0.5ϵ	13,917	9.6	42.6	1,054	0.26	0.46	0.83	0.70	188.0	131.7	83,903.7
Perfect information, 0.7ϵ	13,426	9.9	48.7	817	0.19	0.40	0.88	0.75	213.7	135.5	54,333.4
UCB1+Time Discounting											
Perfect information	59,953	2.2	2.7	1,108	0.37	0.52	0.71	0.61	214.3	129.2	89,638.2
Perfect information, 0.1ϵ	59,953	2.2	2.6	1,136	0.37	0.52	0.71	0.60	214.7	131.0	92,553.4
Perfect information, 0.3ϵ	59,953	2.2	2.6	1,096	0.36	0.51	0.71	0.61	216.7	131.8	87,987.0
Perfect information, 0.5ϵ	59,953	2.2	2.6	1,066	0.36	0.50	0.71	0.61	219.0	130.3	83,275.9
Perfect information, 0.7ϵ	59,953	2.2	3.0	1,038	0.35	0.50	0.71	0.61	222.0	133.0	78,853.9
Best-fitting model	59,953	2.2	4.6	980	0.37	0.50	0.69	0.61	207.7	127.0	80,181.6
Actual	59,946	2.2	3.2	1,063	0.32	0.32	0.76	0.81	209.0	134.0	80,384.7

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D3. All other notes from Table 8 also apply.

Table D16: Simulated Allocations and Outcomes (2005–2015) Under Improved Information and Learning Models, NYSGRC*

	Projects	Trials per project		Unique	% Biomedical		% Novel		Median sequence length		Total
	attempted	Avg	Std	structures	trials	structures	trials	structures	trials	structures	Rewards
Greedy											
Perfect information	8,385	17.8	32.0	2,889	0.77	0.61	0.80	0.77	317.7	311.3	183,254.5
Perfect information, 0.1ϵ	8,682	17.2	34.0	2,797	0.75	0.60	0.80	0.76	317.0	310.7	174,526.0
Perfect information, 0.3ϵ	7,368	20.3	63.0	1,931	0.59	0.49	0.72	0.68	339.0	314.3	100,662.0
Perfect information, 0.5ϵ	6,715	22.2	70.5	1,612	0.55	0.44	0.69	0.65	351.7	316.2	75,784.9
Perfect information, 0.7ϵ	6,452	23.1	73.7	1,445	0.52	0.41	0.67	0.61	356.7	316.0	63,238.5
UCB1+Time Discounting											
Perfect information	59,723	2.5	7.1	1,876	0.70	0.50	0.77	0.68	335.3	310.0	90,836.9
Perfect information, 0.1ϵ	59,723	2.5	7.1	1,888	0.71	0.50	0.77	0.67	336.0	311.2	90,015.6
Perfect information, 0.3ϵ	59,723	2.5	7.1	1,859	0.71	0.50	0.77	0.67	337.0	310.3	88,727.4
Perfect information, 0.5ϵ	59,722	2.5	7.2	1,807	0.71	0.50	0.77	0.67	338.0	309.3	87,144.1
Perfect information, 0.7ϵ	59,723	2.5	7.9	1,728	0.70	0.49	0.78	0.67	338.7	307.3	82,278.9
Best-fitting model	59,733	2.5	9.9	1,363	0.71	0.48	0.78	0.63	336.3	308.7	60,405.2
Actual	59,734	2.5	24.6	1,334	0.75	0.49	0.76	0.66	361.0	319.0	56,485.5

Note: Each simulation uses $\widehat{\boldsymbol{\theta}}$ estimated from the UCB1+Time Discounting model, as reported in Appendix Table D4. All other notes from Table 8 also apply.

Additional References for the Appendices

Buchfink B, Reuter K, Drost HG (2021) Sensitive protein alignments at tree-of-life scale using DIAMOND. *Nature Methods* 18(4):366–368.

Buchfink B, Xie C, Huson DH (2015) Fast and sensitive protein alignment using DIAMOND. *Nature Methods* 12(1):59–60.

Chen X, Christensen TM, Tamer E (2018) Monte Carlo confidence sets for identified sets. *Econometrica* 86(6):1965–2018.

Chernozhukov V, Hong H (2003) An MCMC approach to classical estimation. *Journal of Econometrics* 115(2):293–346.

EMBL-EBI (2021) EMBL-EBI Pfam to PDB mapping. URL `http://ftp.ebi.ac.uk/pub/databases/Pfam/mappings/`, accessed on Mar 13, 2021.

Huang H, McGarvey PB, Suzek BE, Mazumder R, Zhang J, Chen Y, Wu CH (2011) A comprehensive protein-centric ID mapping service for molecular data integration. *Bioinformatics* 27(8):1190–1191.

Kawashima S, Pokarowski P, Pokarowska M, Kolinski A, Katayama T, Kanehisa M (2007) AAindex: Amino acid index database, progress report 2008. *Nucleic Acids Research* 36(suppl_1):D202–D205.

Klausen MS, Jespersen MC, Nielsen H, Jensen KK, Jurtz VI, Sønderby CK, Sommer MOA, Winther O, Nielsen M, Petersen B, et al. (2019) NetSurfP-2.0: Improved prediction of protein structural features by integrated deep learning. *Proteins: Structure, Function, and Bioinformatics* 87(6):520–527.

Miller S, Janin J, Lesk AM, Chothia C (1987) Interior and surface of monomeric proteins. *Journal of Molecular Biology* 196(3):641–656.

NIH (2019) NIGMS funding opportunities search. URL `https://www.nigms.nih.gov/grants/Pages/Funding.aspx?expired=1`, accessed on Feb 29, 2020.

NIH (2021) NIH RePORT advanced projects search. URL `https://reporter.nih.gov/`, accessed on Feb 29, 2020.

UniProt (2021a) Programmatic access—mapping database identifiers. URL `https://www.uniprot.org/help/api_idmapping`, accessed on Apr 20, 2021.

UniProt (2021b) Programmatic access—retrieving individual entries. URL `https://www.uniprot.org/help/api_retrieve_entries`, accessed on April 18, 2021.

UniProt (2021c) Taxonomy—Eukaryota. URL `https://www.uniprot.org/taxonomy/2759`, accessed on Dec 14, 2020.

UniProt (2021d) Taxonomy—Homo sapiens (human). URL `https://www.uniprot.org/taxonomy/9606`, accessed on Dec 14, 2020.

Varadi M, Berrisford J, Deshpande M, Nair SS, Gutmanas A, Armstrong D, Pravda L, Al-Lazikani B, Anyango S, Barton GJ, et al. (2020) PDBe-KB: a community-driven resource for structural and functional annotations. *Nucleic Acids Research* 48(D1):D344–D353.

Wootton JC (1994) Non-globular domains in protein sequences: Automated segmentation using complexity measures. *Computers & Chemistry* 18(3):269–285.

Acknowledgments

We are very grateful to Shane Greenstein, Myrto Kalouptsidi, Robin Lee, Ariel Pakes, and Elie Tamer for their time, patience, and insightful advice. We would also like to thank John Everett, John Norvell, and Peter Preusch for thoughtful discussions on the field of structural biology, the NIH, the Protein Structure Initiative, and the operations of structural biology labs. We thank Shengmao Cao, Varanya Chaubey, Chaim Fershtman, Yannai Gonczarowski, Tianxiao Han, John Hill, Ryan Hill, Charles Hodgson, Louis Kaplow, Max Kasy, Jacqueline Ng Lane, Lucas De Lima, Zhi Lin, Alexander MacKay, Kyle Myers, Frank Pinter, Devesh Raval, Tim Simcoe, Ken Simons, Chris Snyder, Paula Stephan, Jeff Strabone, Senmiao Sun, Wei Yang Tham, Audrey Tiew, Nataliya Langburd Wright, Hanbin Yang, Ron Yang, David Zhang, participants of the Harvard Industrial Organization workshop, WICK#8 Doctoral Workshop, EARIE 2022, NBER Productivity Seminar, Innovation & Entrepreneurship Seminar at Max Planck Institute, MIT Junior Researcher Series Talk, IIOC 2023, NBER Summer Institute 2023 Science of Science Funding session, Economics Research Seminar at Instacart, Zhejiang University School of Economics seminar, Dartmouth IO Conference 2023, INFORMS Annual Meeting 2023, Duke Fuqua Junior Strategy Conference 2023, and Berkeley-Haas Entrepreneurship and Innovation Seminar for helpful suggestions. We are also very grateful to Bill Lovejoy and Izak Duenyas for providing helpful feedback on this revised draft. We thank Bob Freeman and staff at Harvard Research

Computing for technical assistance and the Doctoral Office at Harvard Business School for financial assistance. We have used LLMs to proofread and make stylistic improvements to sentences and paragraphs of this paper. All errors are our own.