Healthcare Mechanisms from Policy-as-Code Search under Strategic Provider Response

Anonymous ACL submission

Abstract

Healthcare mechanisms are inseparable from the strategic provider response they induce: existing healthcare AI benchmarks hold this response fixed and so cannot evaluate mechanisms by the equilibrium they produce. We recast hospital mechanism design as program synthesis for language models: typed, inspectable rule programs are executed and scored by Medi-Sim, a multi-agent simulator with five strategic provider channels (coding, selection, delay, effort, triage). An incentive sweep recovers classical health-economics findings as adjacent regimes—up-coding and low-complexity-patient selection under profit pressure, and Goodhart-style drift where measured performance becomes anti-correlated with true outcomes—and a single audit lever exposes *pressure migration*: closing the coding channel more than doubles low-complexity selection. LLM-guided evolutionary code search over the same rule-program space then synthesizes an inspectable mixed-objective program that eliminates up-coding, halves rejection, and retains most of the profit-oriented baseline's funds.

1 Introduction

A case-based payment becomes a coding rule, an audit reshapes patient selection, and a quality bonus redirects effort toward the measured score: in each case, a hospital mechanism is realized as the *composition* of an administrator's instruction with a provider's best response, and the composition—not the text—determines billing, access, and outcome.[1] The dynamic we focus on is *pressure migration*—a multi-channel feature of strategic best response in which, when a rule closes one provider channel, the same incentive resurfaces in an adjacent one, so a benchmark that scores rules against a fixed provider systematically over-rewards mechanisms whose effect is to relocate rather than remove distortion. We therefore evaluate hospital mechanisms inside a closed-loop strategic-response simulator, and because regulated deployment additionally requires every rule to remain auditable line by line, we restrict the administrator's policy class to inspectable, typed rule programs—reframing mechanism design as *program synthesis* over a constrained administrative interface.

Pressure migration is visible across three decades of healthcare reform. Hospitals respond to Medicare diagnosis-pricing changes by re-coding rather than by treating more patients (Dafny, 2005); Medicare Advantage risk scores grow faster than fee-for-service scores through coding intensity (Kronick and Welch, 2014); and English NHS waiting-time targets change both reported waits and the operations that produce them (Bevan and Hood, 2006; Propper et al., 2010). The machine-learning reading is direct: each is *Goodhart-style drift* (Manheim and Garrabrant, 2018) delivered through a strategic-response shift in the data-generating process, of the kind formalized by strategic classification and performative prediction (Hardt et al., 2016; Perdomo et al., 2020).

Existing benchmarks cannot see this dynamic. Healthcare AI environments train clinician-level policies on a fixed environment with passive providers (Komorowski et al., 2018; Yu et al., 2021; Gottesman et al., 2019), treating provider behavior as exogenous noise. Automated mechanism-design systems do model strategic response, but instantiate

[1] Healthcare-specific terminology used throughout the paper—including hospital DRG and DRG-style arrivals, hospital CMI, hospital KPI and KPI steering, the five provider-response channels (coding, selection, delay, effort, triage), the coding and measurement wedges, gold-plating, skimping, cream-skimming, the Identify–Produce–Settle (IPS) loop, and the hospital policy DSL—is collected in the glossary of Appendix L.

taxes, auctions, or generic allocation rather than healthcare primitives—reimbursement, audits, care-team queues, measured quality—and their searched controllers are black-box neural networks that fail line-by-line auditability (Zheng et al., 2022; Dütting et al., 2024; Sandholm, 2003). Neither camp scores administrator rules and strategic provider responses through realized access, reimbursement, and performance in the same rollout.

We instantiate the missing loop in Medi-Sim. Administrator rules are written as *policy-as-code*: typed, executable expressions over a fixed set of approved levers (incentive coefficients, audit intensity, bonus pool, performance-score weights) that are auditable line by line (Rudin, 2019), and expose the kind of clear, context-relevant information emphasized in good machine learning practice for medical devices (GMLP)(U.S. Food and Drug Administration et al., 2021). Providers respond through five named channels—coding, selection, delay, effort, and triage—drawn from health economics (Ellis, 1998; Ma, 1994; Kuhn and Siciliani, 2008; Holmstrom and Milgrom, 1991), and an Identify–Produce–Settle (IPS) loop keeps rules, responses, and outcomes in the same rollout, treating settlement (reimbursement, scores, bonuses) as part of the mechanism rather than a reporting layer. The same loop is the search interface: because candidates are typed code expressions over a small state-feature set, useful mutations are *semantic* edits over code rather than gradient steps or random rewrites—a regime in which LLM-guided code search outperforms unguided genetic operators (Romera-Paredes et al., 2024; Lehman et al., 2022; Novikov et al., 2025), while line-by-line auditability rules out neural controllers. The language model therefore acts as a code-editing operator over the rule program under a safety-penalized closed-loop fitness; provider agents are parameterized response classes, not LLMs.

Three experiments close the loop. An incentive sweep recovers classical findings as adjacent regimes of one phase diagram—up-coding and low-complexity selection under profit pressure, balanced-interior Goodhart drift—and administrative lever sweeps expose pressure migration: audits shift pressure from coding to selection, while bonus pools and KPI-steered flex capacity reveal proxy and waiting-time failures. LLM-guided code search over the same rule interface refines a diverse warm-start library into an inspectable mixed-objective program that eliminates up-coding, halves rejection, and retains most of the profit-oriented baseline's funds; ablations show warm-start priors and LLM-guided refinement to be jointly necessary.

Contributions. (1) An LLM-program-synthesis testbed for high-stakes mechanism design. We recast provider-side mechanism design as LLM-guided program synthesis over a typed administrative DSL, in which neural controllers are excluded by audit requirements and the LLM acts as a code-editing operator on inspectable rule programs under safety-penalized multi-agent rollouts. **(2) A closed-loop strategic-response benchmark.** We release Medi-Sim, an Identify–Produce–Settle simulator that keeps administrator rules, five strategic provider channels, and realized access/reimbursement/performance in the same rollout, exposing the channel-level diagnostics needed to detect strategic-response distortion. **(3) Pressure migration as a benchmark phenomenon addressable by LLM-guided code search.** Classical healthcare failures occupy adjacent regimes of one mechanism space, and LLM-guided refinement of a diverse warm-start library can reduce targeted manipulation while monitoring whether pressure reappears on adjacent channels; in the main held-out mixed-policy comparison, the searched program closes the coding channel without increasing rejection. Ablations attribute the effect jointly to priors and LLM code editing.

2 Problem Formulation

We model the hospital as a finite-horizon stochastic Stackelberg game over T periods. The *hospital administrator* is the leader, committing to a mechanism action u_t at each step; the *provider population* is the follower, drawn from a tractable response class Π_P described below. Throughout, J indexes care teams.

State and leader action. The hospital state X_t collects funds F_t, congestion Q_t, per-team queues $\{\mathcal{Q}_{j,t}\}$, the previous-period KPI vector, and reputation Rep_t (Eq. (6), App. B). The leader action u_t collects incentive coefficients (α_t, β_t) for provider finan-

cial and quality sensitivity, total and flexible capacities $(B_t^{\text{tot}}, B_t^{\text{flex}})$, the bonus pool B_t^{pool} and softmax sharpness κ, the KPI weights $(w_H, w_W, w_{\text{rej}}, w_C)$ on health/waiting/rejection/cost, audit intensity q_t, and an optional KPI-steering switch ξ_t (Eq. (7), App. B).

Follower action: five distortion channels. Each team j observes its queue, capacity signals, fatigue, and incentives, and chooses an action that decomposes into five channels:

$$a_{j,t} = (\underbrace{\hat{g}_{i,t}}_{\textit{coding}}, \underbrace{d^{\text{acc}}_{ij,t}}_{\textit{selection}}, \underbrace{d^{\text{def}}_{ij,t}}_{\textit{delay}}, \underbrace{E_{ij,t}}_{\textit{effort}}, \underbrace{R_{ij,t}}_{\textit{triage/resource}}), \tag{1}$$

indexed by candidate patient i. These are exactly the five channels through which providers strategically respond to medical mechanisms in the health-economics literature (Ellis, 1998; Ma, 1994; Dafny, 2005; Kuhn and Siciliani, 2008; Holmstrom and Milgrom, 1991), and they map one-to-one to the distortion measurements reported in §5. The five-channel choice covers the main margins exposed by the Identify–Produce–Settle loop without making the response model too broad to diagnose channel-by-channel behavior. The provider-side team utility that drives these five channels is

$$\begin{aligned} U_{j,t} &= \alpha_t(\text{Rev}_{j,t} - C_{j,t}) + \beta_t H_{j,t} \\ &\quad + \theta\,\text{Bonus}_{j,t} - \nu\left[\max(0, \text{Load}_{j,t} - \bar{E}_j)\right]^2, \end{aligned} \tag{2}$$

where $\theta > 0$ is the fixed weight on the realized bonus and $\nu > 0$ is the convex fatigue penalty above per-team load capacity $\bar{E}_j$.

Bounded-rationality response class. We restrict the follower to a tractable response class $\Pi_P = \{\pi_P^\phi : \phi \in \Phi\}$ parameterized by interpretable behavioral coefficients ϕ that govern, per channel, how aggressively the team's action moves with the local gradient of Eq. (2); the functional forms are given in §3. This is a deliberate design choice rather than an equilibrium claim: it preserves per-channel identifiability, keeps each channel inspectable, and matches the comparative-static predictions used to validate the simulator in §5.

Stackelberg objective. A mechanism is evaluated only through the rollout distribution induced by the follower's response. The leader optimizes a chosen social objective $o \in \{\text{welfare}, \text{profit}, \text{mixed}\}$ with per-objective discounted return on seed s,

$$G^o(\pi_A, \pi_P^*; s) = \sum_{t=1}^{T} \gamma_d^{t-1}\, r_A^o(X_t, u_t), \tag{3}$$

and solves

$$\begin{aligned} \pi_A^{o,*} \in \arg\max_{\pi_A \in \Pi_A} \big\{ &\mathbb{E}_{s,\pi_P^*}[G^o] \\ &- \lambda_{\text{unsafe}}\mathbb{E}[V] - \lambda_{\text{var}}\text{Var}_s[G^o] \big\}, \end{aligned} \tag{4}$$

where $\pi_P^* \in \Pi_P(\pi_A)$ is the bounded-rationality best response, V aggregates safety/distortion diagnostics (unsafe waiting, high-complexity deferral, up-coding, rejection, insolvency), and the variance term is a seed-reliability regularizer (App. B). A mechanism is successful only if the induced provider behavior remains acceptable on every diagnostic, on the average seed *and* reliably across seeds. We defer the choice of policy class Π_A to §4, which discharges auditability constraints by instantiating Π_A as a typed inspectable program class and solves Eq. (4) by AlphaEvolve-style code search.

3 The Medi-Sim Environment

Medi-Sim implements the Identify–Produce–Settle (IPS) decomposition of §2 with the policy interface shown in Figure 1: hospital administrative rules are fixed for an episode, clinician-side programs respond within those rules, and the resulting rollout dashboard exposes the same distortion channels used by L1–L3 (Algorithm A.1, App. A gives the period-by-period loop). We describe each primitive in turn.

3.1 Identify: arrivals, classification, and the coding wedge

At each step, a default hospital DRG-style arrival process[2] draws a Poisson batch of patients $\mathcal{P}_t$ with rate λ_{arr} (Eq. (10), App. C); non-Poisson kernels are admissible (App. L). Each patient i carries a true group $g_i^\star \in \mathcal{G}$, a normalized hospital case-mix index CMI[3] value $\text{CMI}_i^\star \in [0,1]$, urgency $\text{Urg}_i \in [0,1]$,

[2]Hospital DRG-style arrivals carry clinical type, urgency, tolerance, and reimbursement-relevant case weight; see Appendix L.

[3]CMI is the normalized hospital case-mix index used as patient complexity and as the payment-relevant weight that coding can distort; see Appendix L.

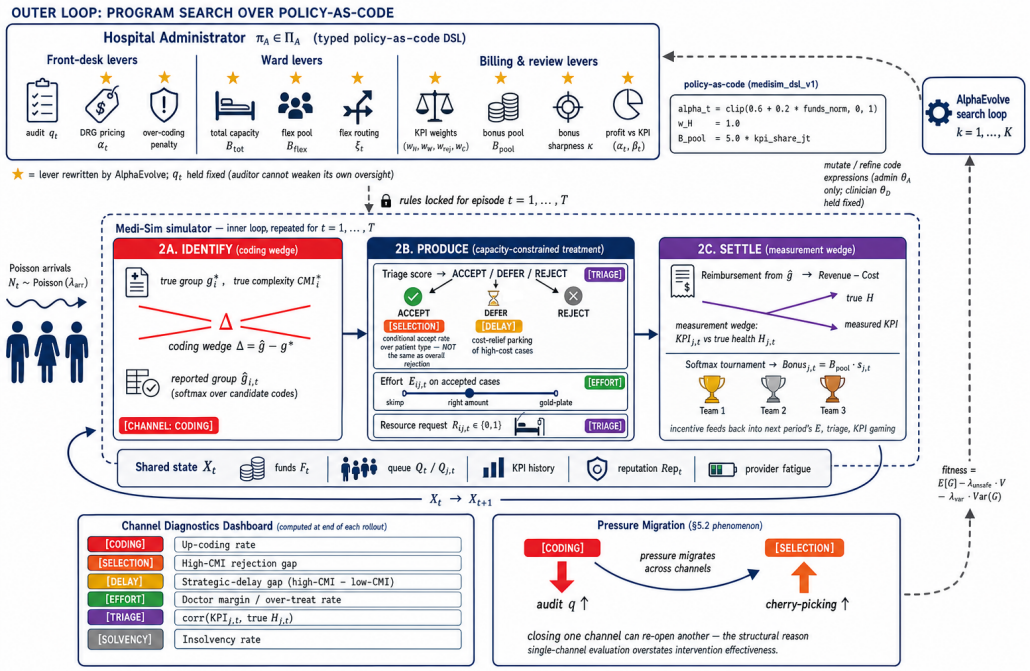


Figure 1: MEDI-SIM IPS and policy-as-code overview. Top: the hospital administrator writes episode-level front-desk, ward, and billing/review rules; stars mark levers refined by ALPHAEVOLVE. Middle: clinician programs respond within locked rules through the Identify–Produce–Settle loop. Bottom: the dashboard reports channel-level diagnostics that guide policy search.

and waiting tolerance. The default simulator uses an aggregated macro-DRG distribution inspired by common inpatient categories and DRG relative weights (Centers for Medicare & Medicaid Services, 2026; Healthcare Cost and Utilization Project, 2025). True complexity determines clinical need; coded complexity determines settlement. This split is the *hospital coding wedge*[4]: it creates the rent that the coding channel arbitrates.

Coding action. A hospital coder chooses a reported group $\hat{g}_{i,t}$ from a finite candidate set around the true group by applying a score-based choice rule to candidate groups. For each candidate group, the host simulator computes candidate-level signals: incremental reimbursement $\Delta R_i(g)$, audit-expected penalty under the configured audit schedule, ethics pressure from coded-complexity inflation, and the resulting coding gap. The coding rule maps these signals to a scalar candidate score; the baseline closed form is Eq. (11) in App. C.

The same score can be instantiated either as a stochastic softmax choice or as its deterministic zero-temperature limit, $\hat{g}_{i,t} = \arg\max_g s_i(g)$. The AlphaEvolve L3 implementation uses the deterministic variant to reduce evaluation noise, while keeping the candidate set, score features, audit schedule, and settlement routine fixed. As incentives vary, this score-based channel produces intermediate aggregate up-coding rates across patients and rollouts rather than making coding an all-or-nothing administrative switch.

In L3, `candidate_score` is only the exposed behavioral scoring map for this channel. Edits to it reweight the coder's sensitivity to fixed host-computed features such as `upcode_pressure`, `audit_penalty`, `ethics_pressure`, and `coding_gap`; candidate construction, audit, penalties, and clawback remain host-side settlement routines.

Patient routing. Patients register to service units rather than being reassigned by the hospital. In L1/L2 active strategic routing is disabled and KPI-aware steering acts only on the flexible capacity pool (§3.3), so selection and delay arise predominantly through the provider triage channel (§3.2); the external-validity implications are discussed in Limitations.

[4]The hospital coding wedge is the gap between true clinical complexity and the reported billing group used for settlement; see Appendix L.

3.2 Produce: capacity-constrained treatment

A service unit observes its queue, capacity signals, fatigue, and incentives. Its action realizes four of the five distortion channels at once: *triage/resource* (accept, reject, defer, request a constrained resource), *selection* (the conditional distribution of acceptance over patient types), *delay* (deferral as a cost-relief lever), and *effort* (intensity per accepted case). All four channels are implemented as closed-form behavioral rules that operate on the local gradient of the team utility $U_{j,t}$ defined in Eq. (2); collectively, these rules are the concrete instantiation of the response class Π_P used in §2.

Treatment production. For an accepted patient i treated by unit j, true health output follows a diminishing-return production in effort $E_{ij,t} \geq 0$, gated by a constrained resource $R_{ij,t} \in \{0, 1\}$ and scaled by team skill and inverse CMI; cost is convex in effort with exponent $\phi > 1$ (Eqs. (12)–(13), App. C). Effort therefore arbitrates the intensive margin (gold-plating versus skimping) while triage and selection arbitrate the extensive margin (who is treated and when), exactly as in the two-margin view of provider response (Ellis, 1998; Ma, 1994).

Triage and delay. Triage gates accept/reject/defer through a scalar score combining urgency, waiting, predicted margin, capacity, fatigue, and KPI-targeting signals; strategic delay is the cost-relief channel that parks high-cost cases under profit pressure, producing the L1 delay signatures of §5.1.

3.3 Settle: Hospital KPI, bonuses, and the measurement wedge

Team-level measured performance combines health with operational and financial penalties:

$$\begin{aligned}\mathrm{KPI}_{j,t} = w_H \bar{H}_{j,t} - w_W \bar{W}_{j,t} \\ - w_{\mathrm{rej}} \mathrm{Reject}_{j,t} - w_C \bar{C}_{j,t}.\end{aligned} \tag{5}$$

Bonuses are allocated through a softmax tournament over $\mathrm{KPI}_{j,t}$ with sharpness κ and pool B_t^{pool}, yielding a local marginal bonus pressure $B_t^{\mathrm{pool}} \kappa\, s_{j,t}(1 - s_{j,t})$ that feeds back into effort, triage, and KPI-targeting behavior (Eqs. (14)–(16), App. C). Because the measured hospital KPI is a weighted aggregate of true health, waiting, rejection, and cost, it is generically misaligned with the principal's objective whenever complex care raises true health but worsens the proxy score—this is the *hospital measurement wedge*[5] that hosts Goodhart-style gaming on triage signals (Holmstrom and Milgrom, 1991; Baker, 1992; Bevan and Hood, 2006). The L1 incentive sweep in §5.1 shows that this wedge becomes large and negative in exactly the intermediate-incentive interior region predicted by multitask theory.

4 Strategic Policy-as-Code

Design desiderata. For regulated healthcare deployment, an admissible policy class Π_A must be (i) *inspectable* line-by-line for compliance review; (ii) *regulable* over a fixed lever set whose semantics match real-world payer and hospital contracts, so that search cannot smuggle in new state variables, measurements, or distortion channels; (iii) *sufficiently expressive* to admit state-conditional rules on observable aggregates (waiting, rejection, profit, utilization); and (iv) *stress-testable* against the response class Π_P of §2. Black-box neural controllers (Zheng et al., 2022; Dütting et al., 2024) satisfy (iii) but neither (i) nor (ii).

Policy class: typed executable programs. We instantiate Π_A as a typed assignment-only DSL. A candidate policy bundle exposes the search-writable administrative expressions: (α, β), total and flexible capacities, the bonus pool and sharpness κ, the KPI weight vector $(w_H, w_W, w_{\mathrm{rej}}, w_C)$, and an optional KPI-steering switch. The audit schedule is not a DSL field: the audit intensity q_t, the audit-hit function p_{audit}, penalty multipliers, and claw-back logic are fixed host-side configuration for a given rollout/evaluation setting. The bundle also includes selected provider-response expressions for effort, triage/resource requests, and coding candidate scoring, which instantiate the bounded-rationality response class Π_P.

L1 and L2 keep the provider-response rules and administrative rule programs fixed at the baseline of §3.2, except for the designated one-at-a-time diagnostic sweeps. L3 allows search over selected provider-response and adminis-

[5] The hospital measurement wedge is the gap between true clinical value and the measured hospital KPI used for bonuses or steering; see Appendix L.

trative expression constants, but the simulator dynamics, patient generation, exposed features, metric computation, host-side clipping, feasibility projection, audit schedule, and settlement/audit routines remain fixed. Thus Π_P remains a stress-test reference class for the simulator's response channels: L3 reweights bounded-rationality responses inside the same provider-response class rather than introducing a new provider model, new measurements, or a co-evolved opponent.

Search and evaluation. We instantiate this policy class with AlphaEvolve-style evolutionary code search (Novikov et al., 2025; Romera-Paredes et al., 2024), implemented through OpenEvolve. A candidate program is first checked for syntactic and type validity, restricted to assignment-only edits over fixed policy fields, and then evaluated on short and full stochastic rollouts; Algorithm A.2 (App. A) summarizes the outer loop. The scalar fitness is the empirical estimator of the Stackelberg objective Eq. (4) with π_P^* taken as the bounded-rationality best response, and the mixed objective uses a log-scaled funds-plus-reputation reward (Eqs. (8)–(9), App. B). Programs that achieve high G through up-coding or unsafe deferral incur a large V and are demoted, so the search trajectory implements the safety constraint rather than relying on a post-hoc filter.

5 Experiments

The experiments follow a two-stage validation-to-discovery logic. We first verified that nine canonical healthcare stylized facts—DRG coding rent, profit-driven case-mix distortion, audit-induced channel substitution, target gaming, queueing capacity response, flexible-capacity allocation effects, quality-side multitasking, risk-adjustment up-coding, and bonus-pool misalignment—reproduce *directionally* as the simulator's incentive, audit, and capacity levers are moved. The full validation map, with anchor citations and per-row signatures, is in Appendix Table 4. With this external check in place, we use Medi-Sim as a mechanism diagnostic across three layers: L1 maps provider responses over the incentive surface; L2 perturbs administrative levers and traces where provider pressure moves; L3 tests policy-as-code search under welfare, profit, and safety-penalized mixed objectives. All L1/L2 summaries report 30-seed means over horizon $T = 200$ unless otherwise stated; Appendix H reports the incentive grid, L2 lever sweeps, defaults, and routing/steering switches.

Empirical findings. The experiments highlight four mechanism-level findings. First, classical healthcare failures occupy neighboring regions of an incentive phase diagram. Second, the balanced interior hides risk on less-visible margins: up-coding and rejection recede while delay and KPI targeting intensify. Third, administrative controls move pressure across channels—audit closes coding but raises selection, bonus expansion worsens proxy alignment, and KPI-steered flexible capacity raises waiting. Fourth, policy-as-code search follows its objective: pure-profit search amplifies coding, whereas the mixed objective reshapes incentive geometry and removes up-coding while preserving much of the return.

5.1 L1: failure regimes form one incentive phase diagram

L1 sweeps financial and quality sensitivities over an 11×11 grid. The response surface is structured but non-monotone: weak incentives ration access; profit pressure activates coding and selection (up-coding 0.226, high-CMI rejection gap 0.182); quality pressure suppresses these channels while increasing effort and solvency stress; and the balanced interior shifts pressure to delay and KPI targeting (representative-regime metrics in Appendix Table 2). The profit- and quality-driven regions recover familiar incentive-theory predictions, but the balanced interior is most diagnostic: visible metrics improve while pressure moves into selective deferral and proxy targeting—high-CMI delay is 0.290 versus 0.010 for low-CMI patients, and measured KPI becomes negatively aligned with true health (-0.659). L1 therefore exposes an interior failure hidden by single-channel diagnostics.

5.2 L2: administrative levers move pressure across channels

L2 reads administrative levers as pressure-tracing interventions: each lever is evaluated by its full response vector across coding, selection,

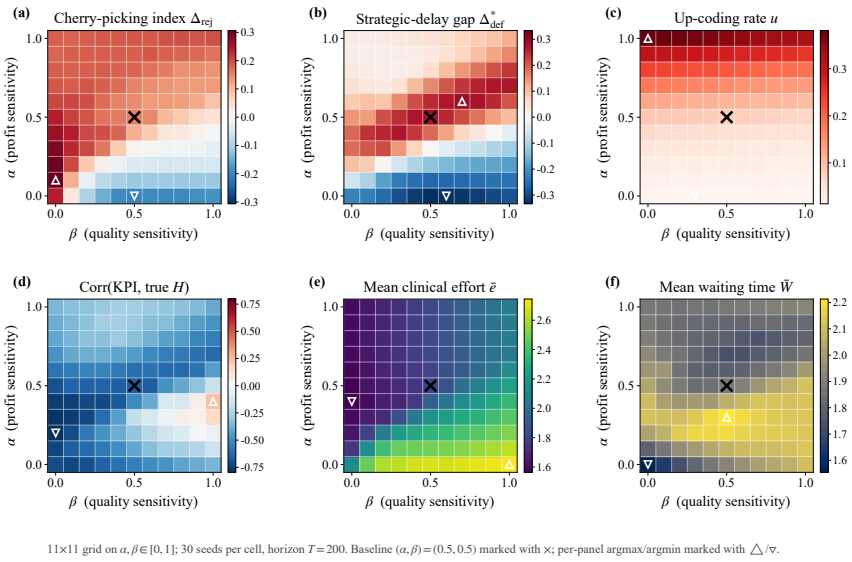


Figure 2: L1 incentive phase diagram. Each panel reports 30-seed means over the 11×11 (α, β) grid. The grid separates low-incentive access rationing, profit-driven coding and selection, quality-driven effort and budget pressure, and balanced-interior delay and KPI targeting.

delay, effort, and triage.

The audit sweep gives the cleanest substitution pattern. Raising audit probability lowers balanced-regime up-coding, while cherry-picking rises. Audit reallocates pressure from billing to selection, making access and delay diagnostics necessary alongside billing accuracy.

The bonus-pool sweep shows the measurement version of the same mechanism. Larger bonuses strengthen the reward attached to the measured KPI. When that proxy is misaligned with true health, stronger incentives widen the wedge: at the endpoints of the balanced-regime sweep, KPI–true-health correlation falls from -0.447 at $B^{\text{pool}} = 0$ to -0.839 at $B^{\text{pool}} = 15$. This is the main Goodhart-style L2 result.

The flexible-capacity sweep gives the operational version: under KPI steering, balanced waiting rises from 1.88 to 2.23 as the flexible pool grows because additional capacity follows bonus-sensitive teams rather than the longest queues, while a steering-off diagnostic flattens the slope to $1.88 \rightarrow 1.88$ (App. G). Flexible capacity acts through its allocation rule, not through capacity volume alone.

5.3 L3: search follows the incentives

L3 uses policy-as-code search to test how objective choice shapes the discovered mechanism family. Candidates are typed edits over the allowed DSL keys only (App. K); implementation details and seed splits are in App. J.

The welfare objective is the positive-control case: AlphaEvolve improves the welfare-family policy along every dimension in Table 1 while keeping up-coding at zero, and lifts doctor margin—search trims excessive effort cost while preserving the gains rewarded by the welfare objective (App. J.2). The profit objective is the reward-hacking diagnostic: AlphaEvolve makes only a small refinement over Greedy-Profit and spends it through the coding channel (up-coding $0.758 \rightarrow 0.807$ alongside higher funds), showing that coding remains an available optimization channel when safety violations carry no penalty.The mixed objective produces the central L3 result. AlphaEvolve keeps return close to the profit-oriented baseline, reduces the violation score from 8.170 to 3.002, halves rejection, and drives up-coding to zero. The mechanism-level change visible in App. J.2 is that the searched policy lowers local bonus pressure while retaining aggregate performance: search therefore changes *which channels* earn the return, rather than improving a scalar score.The warm-start ablation defines the scope of this result. With a neutral-only library, search fails to recover the mixed family. This scopes the policy-as-code interface as a structured refinement tool over meaningful policy priors.

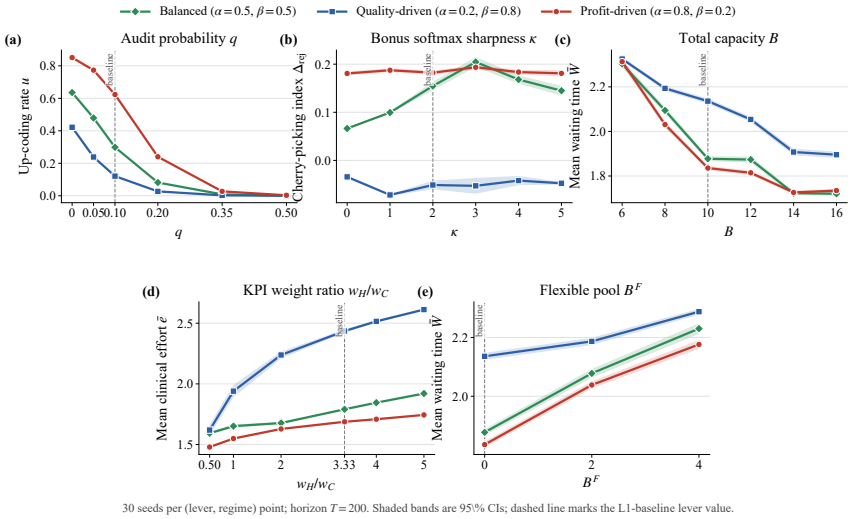


Figure 3: L2 one-at-a-time policy ablations. Curves report 30-seed means over horizon $T = 200$ for balanced, quality-driven, and profit-driven regimes; shaded bands are 95% confidence intervals. Audits suppress up-coding, capacity lowers waiting, and KPI/bonus/flex levers induce nonlinear responses.

Objective	Method	Fitness↑	Wait↓	Reject↓	Upcoding	Funds
Welfare	Greedy-Quality	16.634	1.693	0.032	0.000	10.8
Welfare	AlphaEvolve	**16.932**	**1.529**	**0.011**	0.000	420.6
Profit	Greedy-Profit	121.846	1.784	0.068	0.758	7288.3
Profit	AlphaEvolve	**122.046**	1.786	0.068	0.807	**7353.7**
Mixed	Greedy-Profit	13.580	1.784	0.068	0.758	7288.3
Mixed	Best warm start	13.607	**1.674**	0.058	0.000	5445.9
Mixed	AlphaEvolve	**13.876**	1.727	**0.033**	0.000	5480.7

Table 1: L3 held-out performance. Entries are means over held-out test seeds. Blue cells identify searched policies; green cells mark key comparisons; red flags the profit-only up-coding risk.

6 Discussion and Conclusion

Three classical episodes, one simulator. The episodes motivating §1 map onto distinct (α, β) regions of L1's phase diagram—Dafny-style up-coding at the high-α/low-β corner, Silverman ownership-conditional case-mix distortion along α axis, and Bevan target gaming in the balanced interior. Recovering these patterns under one set of dynamics gives Medi-Sim its benchmark role: a common diagnostic environment for evaluating mechanisms under strategic provider response, rather than a collection of separately tuned stylized examples.

Channel substitution is the structural obstacle. L2 shows that closing one distortion channel can reopen another: audits suppress up-coding, but provider response shifts toward selection or delay, because Π_P's five channels are coupled through Eq. (2). Evaluation must therefore track the whole response, not the single metric a mechanism was designed to improve.

Implications and closing. Medi-Sim makes three evaluation requirements explicit for healthcare policy ML: provider response should be endogenous, searched mechanisms should remain inspectable as code, and scalar reward should be decomposed into channel-level diagnostics so that reward gains through distortion are counted as failures. The released closed-loop environment instantiates these requirements with an Identify–Produce–Settle simulator and an AlphaEvolve-style search interface over a typed DSL. In the held-out mixed-objective evaluation, the searched policy eliminates measured up-coding while retaining much of the profit-oriented baseline's funds, showing that policy-as-code search can reshape the channel through which return is earned. We hope Medi-Sim makes provider-side strategic response a standard benchmark target for ML on healthcare policy.

Limitations

MEDI-SIM is a mechanistic simulator, not a calibrated clinical deployment model, and the design trades external validity for per-channel inspectability in several deliberate ways.

Bounded-rationality response, not equilibrium. The response class Π_P of §2 is implemented as closed-form behavioral rules driven by the local gradient of the team utility (2) rather than as solved equilibria of an inner game. This is a tractable but strict approximation; how well it tracks a fully strategic provider population is an empirical question we do not resolve here.

Aggregated DRG *and patient-choice routing.* The default simulator uses an aggregated macro-DRG distribution and patient-choice registration. Active strategic hospital routing is disabled in L1/L2, and hospital KPI steering acts only on the flexible capacity pool. These choices keep the selection and delay channels attributable to provider triage, which is essential for the L1/L2 attribution claims, but they limit the realism of routing-heavy interventions.

L3 depends on the warm-start library. As reported in §5.3 and Appendix J, the L3 mixed-objective result is a refinement of a diverse warm-start library; with a neutral-only library, $K = 200$ search does not recover the same family. We therefore present AlphaEvolve over MEDI-SIM as a feasibility demonstration of program search over the Mechanism-as-Code policy class rather than as a benchmark-winning algorithm.

Synthetic rollouts. Held-out evaluation uses fixed seed splits over synthetic rollouts; we do not establish real-world effectiveness, and any deployment use would require domain validation, calibration to local case mix, and safety, equity, and legal review.

Ethical Considerations

The simulator studies high-stakes healthcare operations. Its policies must not be interpreted as clinical recommendations or administrative guidance for real hospitals without domain validation, safety review, and fairness analysis. Modeling behaviors such as up-coding, cherry-picking, and strategic delay is intended for detection and stress testing, not for operationalizing manipulation. Any future deployment would require safeguards for patient access, equity across case complexity, clinical safety, privacy, and legal compliance.

References

George P. Baker. 1992. Incentive contracts and performance measurement. *Journal of Political Economy*, 100(3):598–614.

René Bekker, Ger Koole, and Dennis Roubos. 2017. Flexible bed allocations for hospital wards. *Health Care Management Science*, 20(4):453–466.

Gwyn Bevan and Christopher Hood. 2006. What's measured is what matters: Targets and gaming in the english public health care system. *Public Administration*, 84(3):517–538.

Sally C. Brailsford, Paul R. Harper, Bhakti Patel, and Martin Pitt. 2009. An analysis of the academic literature on simulation and modelling in health care. *Journal of Simulation*, 3(3):130–140.

Donald T. Campbell. 1979. Assessing the impact of planned social change. *Evaluation and Program Planning*, 2(1):67–90.

Tugba Cayirli and Emre Veral. 2003. Outpatient scheduling in health care: A review of literature. *Production and Operations Management*, 12(4):519–549.

Centers for Medicare & Medicaid Services. 2026. Acute inpatient prospective payment system. CMS technical documentation and public use files.

Leemore S. Dafny. 2005. How do hospitals respond to price changes? *American Economic Review*, 95(5):1525–1547.

Jinshuo Dong, Aaron Roth, Zachary Schutzman, Bo Waggoner, and Zhiwei Steven Wu. 2018. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation (EC)*, pages 55–70.

Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David C. Parkes, and Sai Srivatsa Ravindranath. 2024. Optimal auctions through deep learning: Advances in differentiable economics. *Journal of the ACM*, 71(1).

Karen Eggleston. 2005. Multitasking and mixed systems for provider payment. *Journal of Health Economics*, 24(1):211–223.

Frank Eijkenaar, Martin Emmert, Manfred Scheppach, and Oliver Schöffski. 2013. Effects of pay for performance in health care: A systematic review of systematic reviews. *Health Policy*, 110(2–3):115–130.

Randall P. Ellis. 1998. Creaming, skimping and dumping: Provider competition on the intensive and extensive margins. *Journal of Health Economics*, 17(5):537–555.

Michael Geruso and Timothy Layton. 2020. Upcoding: Evidence from Medicare on squishy risk adjustment. *Journal of Political Economy*, 128(3):984–1026.

Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. 2019. Guidelines for reinforcement learning in healthcare. *Nature Medicine*, 25(1):16–18.

Linda V. Green. 2002. How many hospital beds? *Inquiry*, 39(4):400–412.

Linda V. Green. 2006. Queueing analysis in healthcare. In Randolph W. Hall, editor, *Patient Flow: Reducing Delay in Healthcare Delivery*, pages 281–307. Springer, Boston, MA.

Moritz Hardt, Nimrod Megiddo, Christos H. Papadimitriou, and Mary Wootters. 2016. Strategic classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science (ITCS)*, pages 111–122.

Healthcare Cost and Utilization Project. 2025. National inpatient sample overview. Agency for Healthcare Research and Quality.

Bengt Holmstrom and Paul Milgrom. 1991. Multitask principal-agent analyses: Incentive contracts, asset ownership, and job design. *The Journal of Law, Economics, and Organization*, 7(Special Issue):24–52.

Seth Karten, Wenzhe Li, Zihan Ding, Samuel Kleiner, Yu Bai, and Chi Jin. 2025. LLM Economist: Large population models and mechanism design in multi-agent generative simulacra. *Preprint*, arXiv:2507.15815.

Matthieu Komorowski, Leo A. Celi, Omar Badawi, Anthony C. Gordon, and A. Aldo Faisal. 2018. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature Medicine*, 24(11):1716–1720.

Richard Kronick and W. Pete Welch. 2014. Measuring coding intensity in the Medicare Advantage program. *Medicare & Medicaid Research Review*, 4(2):E1–E19.

Michael Kuhn and Luigi Siciliani. 2008. Upcoding and optimal auditing in health care (or the economics of DRG creep). CEPR Discussion Paper 6689, Centre for Economic Policy Research.

Joel Lehman, Jonathan Gordon, Shawn Jain, Kamal Ndousse, Cathy Yeh, and Kenneth O. Stanley. 2022. Evolution through large models. *Preprint*, arXiv:2206.08896.

Sagi Levanon and Nir Rosenfeld. 2021. Strategic classification made practical. In *International Conference on Machine Learning (ICML)*, volume 139 of *PMLR*, pages 6243–6253.

Ching-to Albert Ma. 1994. Health care payment systems: Cost and quality incentives. *Journal of Economics & Management Strategy*, 3(1):93–112.

Yecheng Jason Ma, William Liang, Guanzhi Wang, De-An Huang, Osbert Bastani, Dinesh Jayaraman, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2024. Eureka: Human-level reward design via coding large language models. In *International Conference on Learning Representations (ICLR)*.

David Manheim and Scott Garrabrant. 2018. Categorizing variants of Goodhart's law. *arXiv preprint arXiv:1803.04585*.

Alexander Novikov, Ngan Vu, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli, and Matej Balog. 2025. Alphaevolve: A coding agent for scientific and algorithmic discovery. *Preprint*, arXiv:2506.13131.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*.

Juan C. Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. 2020. Performative prediction. In *International Conference on Machine Learning (ICML)*, volume 119 of *PMLR*, pages 7599–7609.

Carol Propper, Matt Sutton, Carolyn Whitnall, and Frank Windmeijer. 2010. Incentives and targets in hospital care: Evidence from a natural experiment. *Journal of Public Economics*, 94(3–4):318–335.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. 2024. Mathematical discoveries from program search with large language models. *Nature*, 625:468–475.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Tuomas Sandholm. 2003. Automated mechanism design: A new application area for search algorithms. In *Principles and Practice of Constraint Programming – CP 2003*, volume 2833 of *Lecture Notes in Computer Science*, pages 19–36. Springer.

Elaine Silverman and Jonathan Skinner. 2004. Medicare upcoding and hospital ownership. *Journal of Health Economics*, 23(2):369–389.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. 2022. Defining and characterizing reward hacking. In *Advances in Neural Information Processing Systems (NeurIPS)*.

U.S. Food and Drug Administration, Health Canada, and Medicines and Healthcare products Regulatory Agency. 2021. Good machine learning practice for medical device development: Guiding principles. Guidance document.

Pieter Van Herck, Delphine De Smedt, Lieven Annemans, Roy Remmen, Meredith B. Rosenthal, and Walter Sermeus. 2010. Systematic review: Effects, design choices, and context of pay-for-performance in health care. *BMC Health Services Research*, 10:247.

Alexander Sasha Vezhnevets, John P. Agapiou, Avia Aharon, Ron Ziv, Jayd Matyas, Edgar A. Duéñez-Guzmán, William A. Cunningham, Simon Osindero, Danny Karmon, and Joel Z. Leibo. 2023. Generative agent-based modeling with actions grounded in physical, social, or digital space using Concordia. *Preprint*, arXiv:2312.03664.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. 2023. Large language models as optimizers. *arXiv preprint arXiv:2309.03409*.

Chao Yu, Jiming Liu, Shamim Nemati, and Guosheng Yin. 2021. Reinforcement learning in healthcare: A survey. *ACM Computing Surveys*, 55(1):1–36.

Stephan Zheng, Alexander Trott, Sunil Srinivasa, Nikhil Naik, Melvin Gruesbeck, David C. Parkes, and Richard Socher. 2022. The AI economist: Taxation policy design via two-level deep multiagent reinforcement learning. *Science Advances*, 8(18):eabk2607.

Appendix Contents

Anonymous code. The anonymized code and reproduction package is available at:

https://anonymous.4open.science/r/Medi_Sim-CC85/

Appendix guide.

- Appendix A: execution and search algorithms.
- Appendix B: formal Stackelberg definitions.
- Appendix C: environment functional forms.
- Appendix D: related work.
- Appendix E: per-channel L1 phase-diagram anatomy.
- Appendices F–G: additional L2 diagnostics.
- Appendix H: L1/L2 setup and hyperparameters.
- Appendix I: external stylized-fact validation.
- Appendix J: L3 search diagnostics and discovered policies.
- Appendix K: implementation details and DSL guardrails.
- Appendix L: terminology and arrival-process note.

A Execution and Search Algorithms

Algorithm A.1 expands the IPS diagram in Figure 1 into the period-by-period simulator loop. Algorithm A.2 gives the outer optimization loop used by L3.

The closed loop contains two feedback channels. Within-rollout feedback is economic and operational: hospital incentives and routing rules shape coding, triage, treatment effort, congestion, settlement, and the next hospital state. Across-rollout feedback is learning-driven: the program-search operator updates the distribution of candidate hospital policies after observing rollout scores and diagnostics, which shifts the strategic and congestion regime explored by later rollouts.

Algorithm A.1 MEDI-SIM simulator execution loop

Require: Hospital capacity and queues, hospital provider teams, hospital administrator policy, reimbursement module, KPI/bonus rule, initial state X_1.

for $t = 1, \ldots, T$ **do**
  Hospital policy step: the hospital administrator sets period levers u_t, including incentives, capacity, bonus-pool, KPI weights, audit intensity, and optional routing or steering parameters.
  Patient generation: sample arrivals $\mathcal{P}_t$ from the configured arrival process and assign latent clinical attributes such as true group, urgency, tolerance, and true complexity $\mathrm{CMI}_i^\star$.
  Identify / coding: hospital coding staff map each arriving patient's clinical presentation to a billable group $\hat{g}_{i,t}$; the gap between true and coded complexity realizes the coding wedge.
  Routing and queueing: register or route patients to hospital care-team queues, carry over backlog, and increment waiting time for patients not served in the current period.
  for each hospital care team j **do**
    Observe local queue $\mathcal{Q}_{j,t}$, capacity, fatigue, incentives, and prior KPI signals.
    Choose provider-side actions: accept/defer/reject, effort $E_{ij,t}$ for accepted cases, and resource request $R_{ij,t}$ under the response class Π_P.
  end for
  Capacity resolution and treatment: enforce hard capacity constraints, treat selected patients, keep unserved patients queued or rejected according to the triage decision, and realize health output $H_{ij,t}$ and cost $C_{ij,t}$.
  Settlement: compute reimbursement from the reported billing group $\hat{g}_{i,t}$, aggregate margin and KPI scores, apply audit penalties, and allocate bonuses.
  State update: update funds, reputation, queues, provider fatigue, KPI history, and the next hospital state X_{t+1}.
end for

Ensure: Rollout trajectory, hospital administrator return, and channel-level diagnostics.

B Formal Definitions for the Stackelberg Formulation

This appendix expands the compact statement of §2.

Hospital state. The state at time t is

$$X_t = \big(F_t,\ Q_t,\ \{\mathcal{Q}_{j,t}\}_{j=1}^{J},\ \mathrm{KPI}_{t-1},\ \mathrm{Rep}_t\big), \quad (6)$$

where F_t denotes funds, Q_t summarizes systemwide congestion, $\mathcal{Q}_{j,t}$ is the queue registered to team j, KPI_{t-1} is the previous-period measured performance vector, and Rep_t is public reputation. The full per-patient state, the patient stream, and the audit signal are observable through these summaries and the exposed state features described in §3.

Leader action. The hospital administrator's mechanism action is

$$u_t = \big(\alpha_t, \beta_t, B_t^{\mathrm{tot}}, B_t^{\mathrm{flex}}, B_t^{\mathrm{pool}}, w_H, w_W, w_{\mathrm{rej}}, w_C, \kappa, q_t, \xi_t\big), \quad (7)$$

with the per-component semantics given in §2. The flexible capacity pool B_t^{flex} is reallocatable

Algorithm A.2 Closed-loop policy search over MEDI-SIM rollouts

Require: Initial policy library $\mathcal{L}_0$, program-search operator, validation seeds, held-out test seeds, rollout horizon T.

1: **for** $k = 1, \ldots, K$ **do**
2: **Deploy:** select or propose a typed hospital policy program $\pi_A^{(k)}$ over the allowed DSL fields.
3: **Simulate:** evaluate $\pi_A^{(k)}$ by running Algorithm A.1 on stochastic validation rollouts.
4: **Score:** compute scalar return G, safety/distortion penalty V, variance penalty, and diagnostic metrics.
5: **Update:** retain or mutate policy programs according to validation fitness and the search operator.
6: **end for**
7: Select the best validation candidate and evaluate it once on held-out seeds.

Ensure: Searched policy, held-out rollout profile, and search diagnostics.

across care teams, and KPI steering through ξ_t assigns that capacity using measured performance scores; see Appendix L.

Safety/distortion penalty. The penalty term V in Eq. (4) aggregates per-step penalties for unsafe waiting, excessive high-complexity deferral, up-coding, rejection, and insolvency; explicit weights are listed in App. K.

Empirical fitness estimator. The search fitness used in §4 is the empirical estimator of Eq. (4),

$$\begin{aligned}\text{Fitness}(\pi) = \mathbb{E}_s[G(\pi; s)] - \lambda_{\text{unsafe}} V(\pi) \\ - \lambda_{\text{var}} \text{Var}_s(G(\pi; s)),\end{aligned} \tag{8}$$

where $G(\pi; s) \equiv G^o(\pi_A, \pi_P^*; s)$ instantiates the per-objective discounted return of Eq. (3) with $\pi_P^* \in \Pi_P(\pi_A)$ taken as the bounded-rationality best response.

Mixed-objective per-step reward. For the mixed objective, one-step reward log-scales funds before combining them with reputation:

$$r_t^{\text{mix}} = 0.5 \cdot \frac{\log(F_t)}{10} + 0.5 \cdot \text{Rep}_t. \tag{9}$$

This return enters the search fitness only and is not interpreted as a standalone welfare or economic-value measure.

C Environment Functional Forms

This appendix collects the closed-form expressions referenced in §3.

Arrival process. The default arrival process is homogeneous Poisson,

$$N_t = |\mathcal{P}_t| \sim \text{Poisson}(\lambda_{\text{arr}}), \tag{10}$$

with per-patient attributes drawn as described in §3.1. Appendix L states when non-Poisson kernels can be substituted.

Coding candidate score. For candidate reported group g, let $\Delta R_i(g)$ be incremental reimbursement and let $\Delta c_i(g)$ be coded-complexity inflation. The candidate score is

$$\begin{aligned}s_i(g) = \alpha_t \, \gamma_{\text{code}} \, \Delta R_i(g) \\ - p_{\text{audit}}(\Delta c_i(g)) \, \lambda_{\text{pen}} \, [\Delta R_i(g)]_+ \\ - r_0 \, \eta_{\text{eth}} \, [\Delta c_i(g)]_+,\end{aligned} \tag{11}$$

where $p_{\text{audit}}(\cdot; q_t)$ is the host-defined audit-hit function under the configured audit intensity q_t. The coded group is chosen by the configured score-based rule: a softmax over $\{s_i(g)\}$ in stochastic response mode, and its zero-temperature $\arg\max_g s_i(g)$ variant in deterministic L3 search/evaluation. Both variants use the same candidate score; intermediate aggregate up-coding arises from patient heterogeneity and incentive-dependent score comparisons rather than from a separate coding mechanism.

Treatment production and cost. For an accepted patient i treated by unit j, true health output follows a diminishing-return production,

$$H_{ij,t} = \text{Skill}_j \left(1 - \exp\left[-\lambda \frac{E_{ij,t}}{\text{CMI}_i^\star + \epsilon}\right]\right) R_{ij,t}, \tag{12}$$

with effort $E_{ij,t} \geq 0$ and resource indicator $R_{ij,t} \in \{0, 1\}$. Cost is convex in effort,

$$C_{ij,t} = C_{\text{fixed}} \, R_{ij,t} + \omega \, E_{ij,t}^{\phi}, \qquad \phi > 1. \tag{13}$$

Bonus tournament and marginal pressure. Bonuses are allocated through a softmax tournament with sharpness κ and pool B_t^{pool},

$$s_{j,t} = \frac{\exp(\kappa \, \text{KPI}_{j,t})}{\sum_{\ell=1}^{J} \exp(\kappa \, \text{KPI}_{\ell,t})}, \tag{14}$$

$$\text{Bonus}_{j,t} = B_t^{\text{pool}} \, s_{j,t}, \tag{15}$$

with local marginal bonus pressure

$$\frac{\partial \text{Bonus}_{j,t}}{\partial \text{KPI}_{j,t}} = B_t^{\text{pool}} \, \kappa \, s_{j,t}(1 - s_{j,t}), \tag{16}$$

which feeds back into effort, triage, and KPI-targeting behavior.

D Related Work

AI and Reinforcement Learning for Clinical Decision-Making. A substantial line of work treats healthcare as a sequential decision problem at the patient level, learning treatment policies from electronic health records under offline RL formulations. The AI Clinician learns vasopressor and fluid policies for sepsis from MIMIC-III and reports lower in-hospital mortality relative to observed clinician behavior (Komorowski et al., 2018), and broader surveys catalogue similar formulations for chronic disease management, anesthesia, and ventilation (Yu et al., 2021). Methodological guidelines emphasize off-policy evaluation, distributional shift, and reward specification as the core obstacles to safe clinical deployment (Gottesman et al., 2019). This literature optimizes the *clinician's* action on a single patient and assumes a fixed reward and a fixed environment. Medi-Sim operates one level above this layer: the action is a hospital administrative rule (incentive coefficients, audit probability, capacity allocation, hospital KPI weights), the environment is a population of strategic providers, and the realized policy is the composition of the hospital administrator's mechanism with the providers' best response. Patient-level clinical RL is thus complementary to, not a substitute for, the mechanism-level question we study.

Healthcare Incentives and Operations. Provider behavior under payment rules has been studied for three decades through agency theory and operational flow. Cost and quality incentives under prospective payment can produce creaming, skimping, dumping, and intensity shifts (Ellis, 1998; Ma, 1994; Eggleston, 2005). Empirical studies on Diagnosis-Related Groups confirm that administrative pricing shifts directly alter case mix and coding (Dafny, 2005; Silverman and Skinner, 2004), and more recent evidence from Medicare Advantage shows that risk-adjusted diagnosis-based reimbursement raises reported risk scores by 6–16% without commensurate change in underlying morbidity (Geruso and Layton, 2020). On the operational side, hospital flow is highly sensitive to capacity: queueing analyses show wait times escalate non-linearly near full utilization (Green, 2002, 2006; Cayirli and Veral, 2003), and flexible bed allocations partially mitigate the load under realistic routing constraints (Bekker et al., 2017). Healthcare-specific simulation has been used for decades to study these flows, but predominantly as isolated discrete-event models without strategic agents (Brailsford et al., 2009). Medi-Sim integrates this macroeconomic payment layer and microeconomic operational layer into a single executable closed loop in which coding, capacity, and revenue interact.

Measurement, Auditing, and Strategic Gaming. Prospective payment introduces an informational wedge between ground-truth clinical complexity and reported billing groups. Pervasive up-coding has been documented across ownership structures (Silverman and Skinner, 2004; Geruso and Layton, 2020), and economic audit theory characterizes how monitoring rates and penalties govern this "DRG creep" (Kuhn and Siciliani, 2008). Performance measurement creates a parallel friction: when scored metrics are imperfect proxies for the planner's objective, high-powered incentives intensify multitasking distortions and target gaming (Holmstrom and Milgrom, 1991; Baker, 1992; Bevan and Hood, 2006). Systematic reviews of pay-for-performance in healthcare reach a consistent verdict that contract design and local context determine whether bonuses improve care or merely move metrics (Eijkenaar et al., 2013; Van Herck et al., 2010). Formal taxonomies of Goodhart-style failures (regressional, extremal, causal, adversarial) clarify why such gaming is structural (Manheim and Garrabrant, 2018). Medi-Sim models the hospital coding wedge and the hospital measurement wedge as dynamic, interactive processes, attributing each distortion to the agent that actually makes the choice (clinician/hospital coder for up-coding, triage for selection and delay, hospital administrator for routing and incentive design), so that channel-level distortion becomes a measurable property of the rollout.

Strategic Machine Learning and Algorithmic Mechanism Design. Our problem interfaces directly with the machine-learning literature on agents that respond to a deployed rule. *Strategic classification* formalizes Stack-

elberg learning against a manipulable agent (Hardt et al., 2016) and has been extended to revealed-preference observations (Dong et al., 2018) and to practical, end-to-end differentiable training pipelines (Levanon and Rosenfeld, 2021). *Performative prediction* generalizes this to settings in which the deployed model itself shifts the data distribution and characterizes performative equilibria (Perdomo et al., 2020). The reward-shaping side of this literature gives formal notions of *reward hacking*: optimizing an imperfect proxy can never be made safe by narrowing the reward function alone under mild structural assumptions (Skalse et al., 2022). In parallel, differentiable economics has used deep architectures to construct optimal multi-dimensional auctions from data (Dütting et al., 2024). These neural mechanisms achieve strong objective values but yield black-box controllers whose parameters resist verification. MEDI-SIM adopts the strategic-learning stance (provider best response is part of the environment) and the algorithmic-mechanism-design objective (synthesize a rule that is robust to that response), but constrains the policy class to read-and-comply programs, which is what regulated healthcare deployment actually demands.

Multi-Agent Simulations and Program-Space Policy Search. Designing macro-policies by modeling micro-agent adaptation has recently scaled through multi-agent reinforcement learning and large-language-model agents. The AI Economist frames optimal taxation as a two-level RL problem in which agents and a planner co-adapt (Zheng et al., 2022). LLM-based generative agents extend this paradigm to believable populations of social actors (Park et al., 2023), and frameworks such as Concordia operationalize generative agent-based modeling with grounded actions and an explicit game-master adjudication layer (Vezhnevets et al., 2023). The LLM-economist line scales mechanism design to large generative simulacra of taxpayers and consumers (Karten et al., 2025). However, these systems instantiate consumption, taxation, or social interaction as their domain primitives, not capacity-constrained clinical care, and they typically train neural controllers whose decision logic is not directly auditable. To navigate a non-differentiable and tightly regulated mechanism space without sacrificing inspectability, we build on LLM-guided evolutionary program search: FunSearch surpassed best-known constructions in extremal combinatorics and online bin packing by treating an LLM as a structured mutation operator over code (Romera-Paredes et al., 2024); ELM showed that LLM-based mutation can drive open-ended evolution in code-defined domains (Lehman et al., 2022); AlphaEvolve generalizes this to algorithmic discovery across mathematics, hardware, and learning systems (Novikov et al., 2025); and Eureka demonstrates that the same recipe can author RL reward functions that outperform expert humans on a majority of robotics tasks (Ma et al., 2024). More broadly, LLMs have been studied as black-box optimizers in their own right (Yang et al., 2023). MEDI-SIM adapts this paradigm to healthcare mechanism design: a candidate hospital policy is a typed executable program over a constrained hospital policy DSL of administrative levers, and the evaluator scores it on stochastic multi-agent rollouts under safety, access, and distortion constraints, yielding policies that are simultaneously high-performing, robust to provider best response, and human-readable.

E A Per-Channel Anatomy of the L1 Phase Diagram

The headline phase diagram of §5.1 reports six top-line outcomes on the 11×11 (α, β) grid. Each of those outcomes, however, is a slice through the joint response of five behavioral channels, and reading any single slice in isolation can mislead. The detailed panels in this appendix, including access decomposition, strategic-delay decomposition, clinical budget panels, and team-level KPI panels, decompose each headline outcome into its constituent primitives. Two structural facts emerge that are not visible at the headline resolution.

First, the four-regime story is preserved at higher resolution: every additional panel either localizes a known regime more sharply or reveals a previously hidden margin without contradicting the headline.

Second, the channels are arranged in a *substitution lattice*: in regions where one distortion saturates against a structural floor or ceiling,

Regime	α	β	Δ rej.	High rej.	Low rej.	Upcode	Effort	Funds
Low-incentive	0.0	0.0	0.241	0.542	0.301	0.014	2.128	23.4
Profit-driven	0.8	0.2	0.182	0.182	0.000	0.226	1.688	**5451.7**
Quality-driven	0.2	0.8	-0.051	0.026	0.078	0.027	2.435	15.2
Balanced	0.5	0.5	0.155	0.157	0.002	0.080	1.790	3486.6

Table 2: Representative L1 regimes. Entries are 30-seed means over horizon $T = 200$; Δ rej. denotes high-CMI minus low-CMI rejection. Colored cells mark regime-defining extrema among the four rows.

an adjacent channel becomes active to relieve the same shadow price. This lattice is the structural reason the headline regimes are separable and is the strongest piece of evidence we have that the IPS decomposition of §3 is the right factorization for Medi-Sim.

E.1 Reading the annotated phase diagram

This subsection provides cell-level annotations (Figure 4) for the L1 grid (Figure 2), serving as the reference for the detailed decompositions that follow. Three patterns are immediately visible. *(i)* The cherry-picking index Δ_{rej} (panel a) attains its maximum at $(\alpha, \beta) = (0.1, 0.0)$ with value 0.31 and inverts in the low-α/high-β band, reaching -0.20 at $(0.0, 0.5)$. The transition generally weakens as quality weight rises, but it is not strictly monotone cell by cell. *(ii)* The strategic-delay gap Δ^*_{def} (panel b) is *not* monotone in either variable: its maximum is 0.29 at $(0.6, 0.7)$, well inside the mixed-incentive corridor, and the panel contains a negative stripe at high β, low α where deferral protects *low*-complexity cases. This pattern is decomposed in the dedicated delay analysis of §E.3. *(iii)* The up-coding rate u (panel c) is monotone in α at every β: values run from about 0.01 at low α to 0.38 at $\alpha = 1.0$. Variation along β at fixed α is much smaller but not zero, reaching about 0.06 in the highest-α row. Coding is therefore the most nearly one-dimensional channel in the parameter pair (α, β), a fact that matters when we discuss the audit-coding-selection substitution in §E.5. The remaining panels, $\text{Corr}(\text{KPI}, H)$, mean effort, and mean waiting, are each interpreted in the dedicated subsections below.

E.2 The access channel: rejection is only half the story

Cherry-picking is conventionally measured as the gap between high-complexity and low-complexity rejection rates, and on that summary statistic alone (Figure 4(a)) the profit-driven corner looks unambiguous. The decomposition in Figure 5 reveals that this headline hides a subtler structural fact.

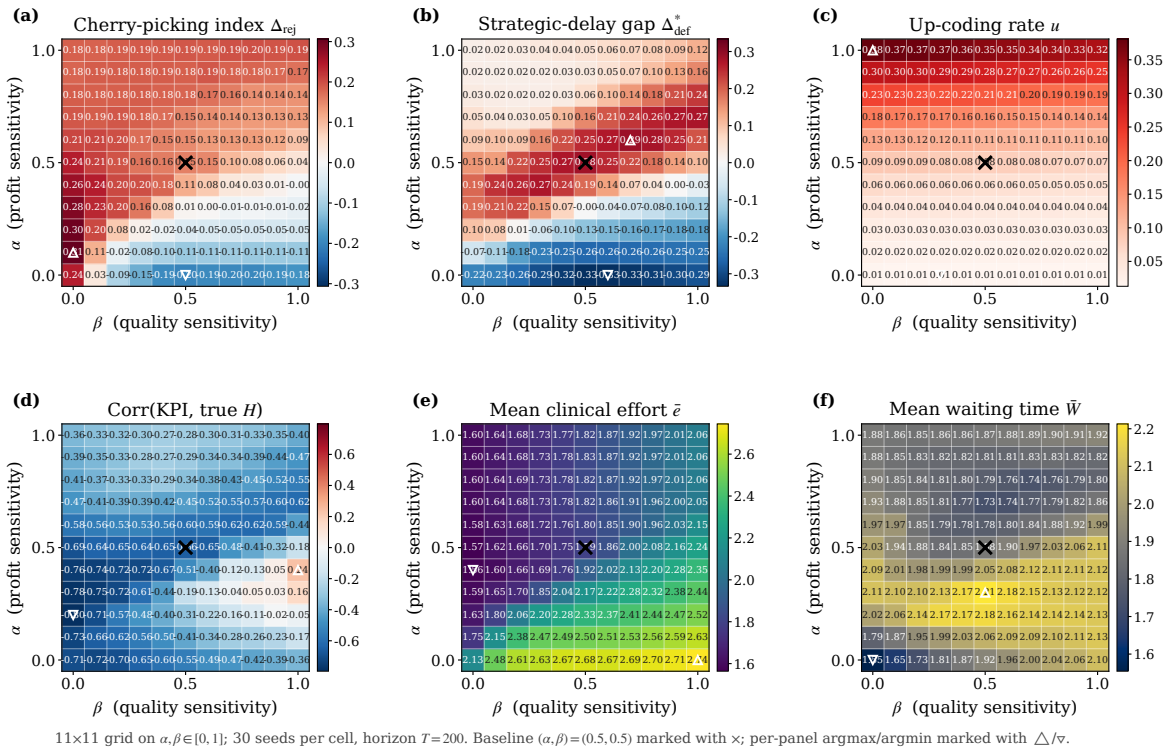
At low α, both rejection rates are high in absolute terms. The providers lack a positive margin and treat acceptance as a pure fatigue cost, so the gap between them is small (Figure 5, panels a–b). As α rises into the profit corner, both rates fall because throughput becomes valuable, but low-CMI rejection falls faster: cheap cases are unconditionally profitable. The gap therefore widens mainly because they accept simple ones more eagerly. The conventional cherry-picking index captures the *size* of the gap but misattributes its *source*.

The deferral panels (d) and (e) tell the complementary story. Overall deferral is not monotone in α; the more stable signal is the high-CMI minus low-CMI deferral gap. That gap becomes positive across the moderate- and high-α band, reaching 0.316 at $(0.6, 0.7)$, while low-α/high-β cells reverse sign. This is a substitution pattern: when outright rejection becomes costly, providers shift cost-shedding into the delay channel, targeting the same patient subpopulation. Panel (f) confirms that this substitution is not free: realized waiting peaks at $(\alpha, \beta) = (0.3, 0.5)$, in the same moderate-incentive corridor where access pressure is redistributed across rejection and deferral.

The headline rejection gap heatmap suggests that the profit corner is the single worst access regime. Figure 5 reveals two distinct worst-case access regimes. Between them, a moderate-incentive corridor produces queue-mediated congestion as rejection and deferral trade off against each other. This third failure mode is invisible to audit-only interventions.This micro-level structure underlies the channel substitution results of §5.2.

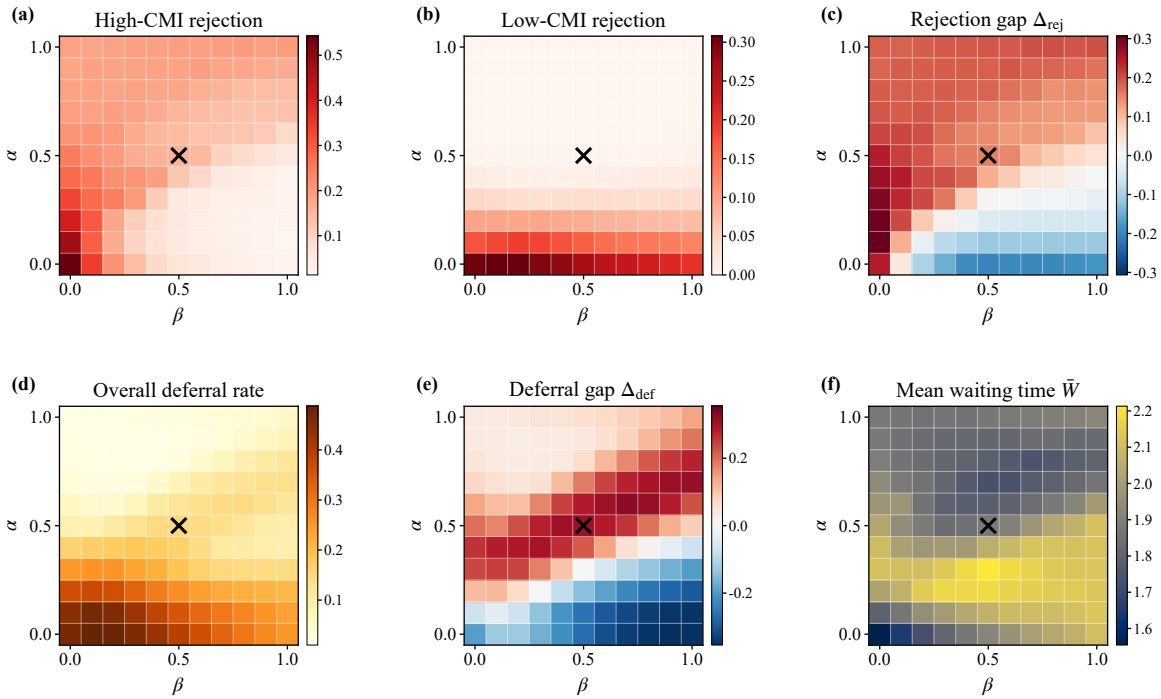
E.3 Strategic delay as a cost-relief substitute

Figure 6 isolates the delay channel from queue-mediated waiting and from involuntary deferral. The four-panel block (d)–(g) restricts the deferral sample to cases where the doctor chooses to defer *despite available local bed capacity*; this is the operationalization of "strategic delay" used in the main text. Three findings emerge.



11×11 grid on $\alpha, \beta \in [0, 1]$; 30 seeds per cell, horizon $T = 200$. Baseline $(\alpha, \beta) = (0.5, 0.5)$ marked with ×; per-panel argmax/argmin marked with △/▽.

Figure 4: Cell-annotated L1 phase diagram. Each cell reports the 30-seed mean of the indicated outcome on the 11×11 (α, β) grid at horizon $T = 200$; × marks the balanced baseline and △/▽ mark the per-panel argmax/argmin. The numerical annotations make it possible to verify the regime boundaries discussed in §E.2–§E.5 cell by cell.



11×11 grid on $\alpha, \beta \in [0, 1]$; 30 seeds per cell, horizon $T = 200$. Baseline (0.5, 0.5) marked with ×.

Figure 5: Access decomposition on the L1 grid. Panels (a) and (b) split rejection by true complexity into high-CMI and low-CMI components, panel (c) is the resulting cherry-picking gap, panels (d) and (e) do the same decomposition for deferral, and panel (f) reproduces the realized mean waiting time. The contrast between (a,b) and (d,e) is the central content of this appendix: as α rises, the access burden migrates from rejection to deferral, but only for high-complexity patients.

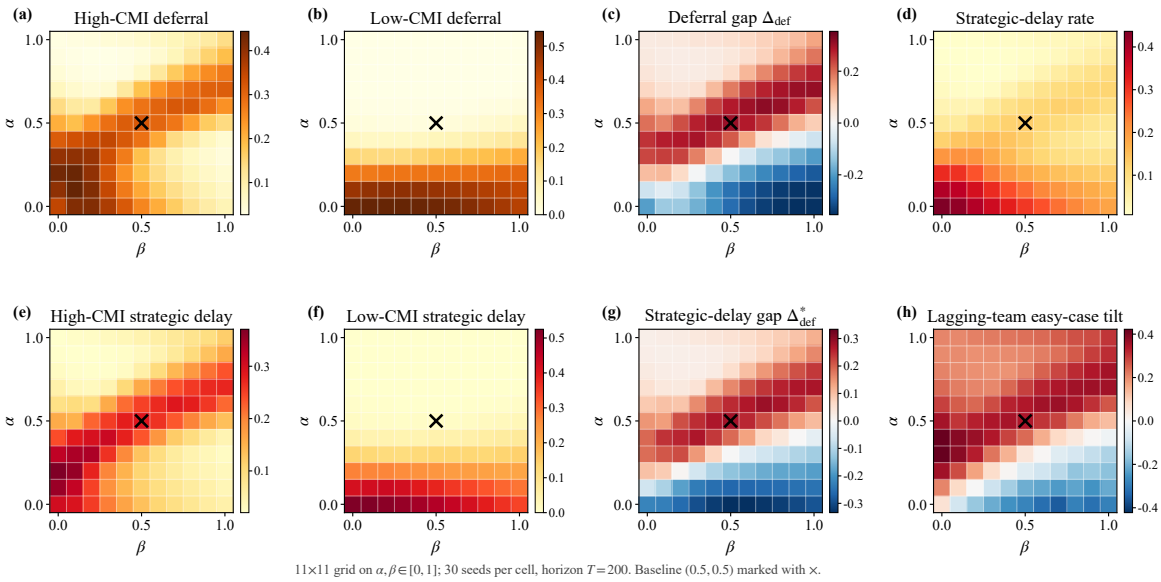


Figure 6: Strategic-delay decomposition on the L1 grid. Panels (a) and (b) report deferral rates separately for high-CMI and low-CMI patients; panel (c) is the deferral gap. Panels (d)–(f) restrict the deferral sample to delays that improve next-period team utility under Eq. (2) ("strategic delay"); panel (g) is the strategic-delay gap; panel (h) is the easy-case CMI tilt of KPI-lagging teams.

First, strategic delay is not concentrated on high-complexity patients everywhere. At low α, low-CMI strategic delay can exceed high-CMI strategic delay, producing the negative gaps in panel (g). In the mixed-incentive corridor the sign flips: high-CMI strategic delay dominates, consistent with the cost-relief motive in the microfoundation.

Second, the strategic-delay gap (g) is the most clearly interior pattern on the grid. Its maximum 0.29 sits at $(0.6, 0.7)$, where neither profit-corner rejection nor quality-corner effort fully dominates and delay becomes the cheapest cost-relief lever. Its minimum -0.33 sits at $(0.0, 0.6)$, where teams protect low-complexity throughput against the cost weight in KPI. Neither pattern is observable from the headline rejection heatmap alone.

Third, the lagging-team easy-case tilt (h) ranges from about -0.35 to $+0.42$ and captures a different mechanism from the strategic-delay gap. The tilt is a *team-level* Goodhart signature: teams trailing in the KPI tournament shift toward easier cases to recover bonus share. This mode becomes salient when bonus pressure $B^{\text{pool}}\kappa\, s_j(1 - s_j)$ is large (see Figure 9 for the corresponding sensitivity).

Taken together, panels (g) and (h) furnish quantitative evidence for the multitasking claim in §5.1: where the headline KPI–health correlation is most negative, the responsible channels are not effort distortion but patient-level delay and team-level case-mix tilt acting in concert.

E.4 Clinical-budget frontier: bounded health, unbounded cost

The dominant qualitative feature of Figure 7 is a scale asymmetry between clinical gains and financial exposure. Mean realized health (panel d) varies only in the narrow band $[0.95, 1.02]$, while terminal funds (panel g) span more than four orders of magnitude. Mean effort (panel a) ranges from 1.56 to 2.74, mean cost (panel e) from 8.23 to 16.41, and health-per-cost efficiency (panel f) collapses from about 0.12 to 0.06 as β rises. The diminishing return treatment production function of §3.2 guarantees that health saturates while cost is convex in effort. The quality-driven corner therefore becomes insolvent because it keeps buying effort on a saturated-health, high-cost segment of the production curve.

Another two secondary patterns are important for interpretation. The effort-per-CMI panel (b) reveals that gold-plating intensifies along β even when the true clinical need is held constant in the denominator, ruling out a pure case-mix explanation. The effort-per-health panel (c) is essentially a mirror image

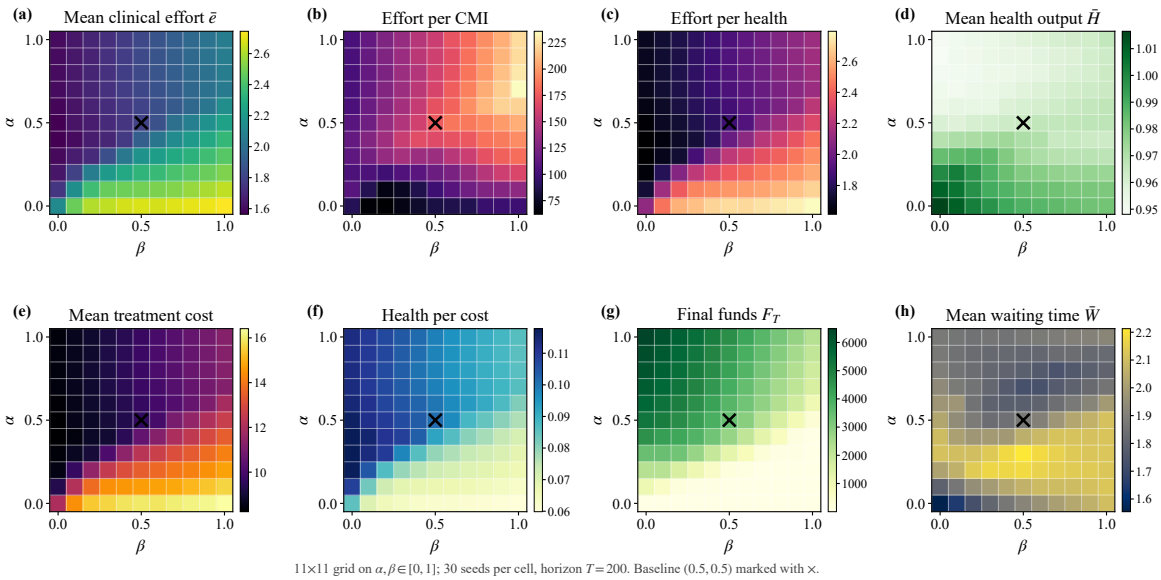


Figure 7: Clinical and budgetary outcomes on the L1 grid. Top row: mean clinical effort, effort per unit true CMI, and effort per unit realized health. Middle row: mean realized health, mean treatment cost, and health per cost. Bottom row: terminal funds F_T and mean waiting time. The contrast between the narrow range of health (0.95–1.02) and the multi-order-of-magnitude range of terminal funds (0.4–6490) is the structural reason why interpreting health in isolation is misleading on this grid.

of the health-per-cost panel (f), with the small but real implication that the marginal product of effort drops by a factor of 1.5–2× as we cross the saturating region; a mechanism designer that uses health as a target without dividing by cost will keep paying for an effort margin that produces almost no measurable health benefit. The waiting panel (h) reproduces the U-shape already seen in Figure 4(f) but at higher resolution: waiting peaks at 2.21 near $(\alpha, \beta) = (0.3, 0.5)$, where selection has not thinned the queue fully.

E.5 KPI proxy management at the team level

Figure 8 is the team-level counterpart to the patient-level decomposition above. Two panels carry most of the interpretive weight.

Panel (a) plots $\mathrm{Corr}(\mathrm{KPI}_{j,t}, H_{j,t})$ across teams and time on each (α, β) cell. The correlation is sharply negative in the low-β band with low or moderate α (reaching -0.80 at $(0.2, 0.0)$), is close to zero along a diagonal corridor through the interior, and becomes positive only in a high-β corridor (peaking at $+0.24$ at $(0.4, 1.0)$). The bonus-health correlation in panel (b) repeats this pattern with sharper amplitude because the softmax tournament amplifies any cross-team ordering present in KPI.

Panel (d), the mean accepted CMI of KPI-lagging teams, can be interpreted as conditional workload level. It ranges from about 0.20 to 0.62 and records the case mix that lagging teams actually end up carrying. Panel (e) reports the lagging-team easy-case tilt, an acceptance-rate gap between low- and high-CMI patients that spans roughly $[-0.35, +0.42]$. This distinction matters in the low-α/high-β corner: lagging teams have high accepted CMI there because quality pressure protects clinically complex patients while weak profit pressure removes the incentive to shed them. The negative tilt in the same region confirms that these teams are not padding KPI with easy cases, but carrying a heavier high-CMI clinical burden. Conversely, positive values in panel (e) identify KPI-driven easy-case selection among lagging teams, a team-level pattern that complements the aggregate rejection-gap view in Figure 4(a).

Panels (f)–(h) are auxiliary diagnostics. We report cross-team dispersion in load, accepted case mix, and outcomes to check whether the patient-level distortions documented above also leave a team-level footprint. The clearest signal is the accepted-CMI variance, which rises in regions where KPI-lagging teams tilt toward easier cases. The bottom row also separates

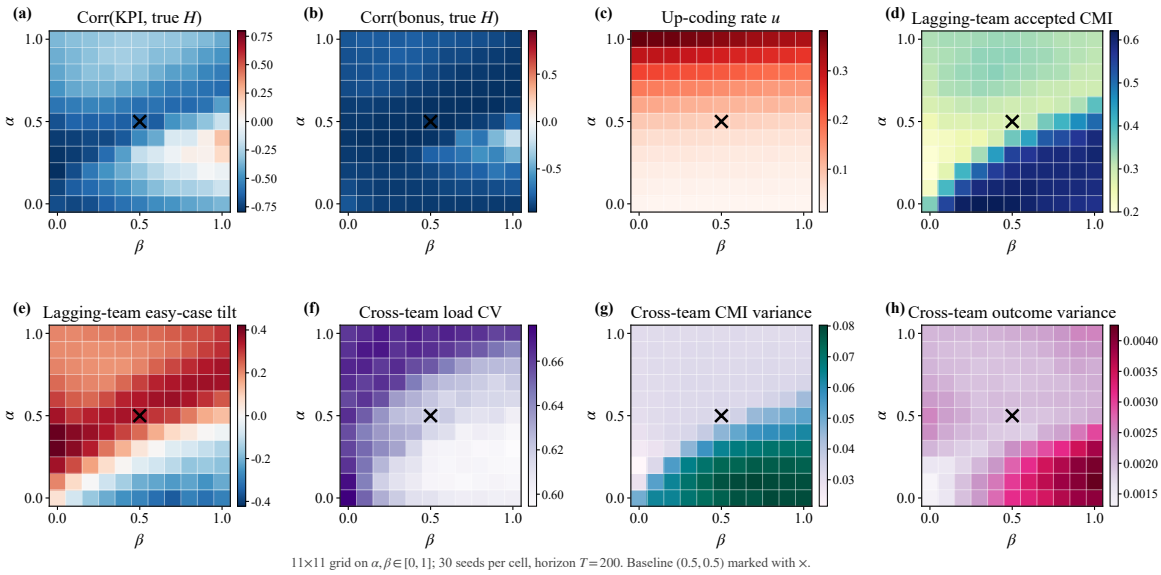


Figure 8: Team-level and KPI-level diagnostics on the L1 grid. Panels (a) and (b) report Pearson correlations of measured KPI and realized bonus against true health, on the cross-team panel. Panels (c)–(e) report the up-coding rate, mean accepted CMI by lagging teams, and the easy-case tilt of lagging teams. Panels (f)–(h) report cross-team coefficient of variation of load, of accepted CMI, and of clinical outcome.

two qualitatively different dispersion regimes. In the low-β band, weak quality pressure makes case-mix sorting the dominant source of cross-team heterogeneity: teams differ mainly in which patients they accept. In the low-α/high-β corner, by contrast, profit-driven shedding is weak and quality pressure protects high-CMI patients, so cross-team dispersion reflects uneven clinical burden, effort intensity, and outcome variation under a saturated production function.

F L2 Additional Lever Diagnostics

The main text reports the three L2 mechanism findings. The remaining one-at-a-time sweeps serve as sanity checks and mechanism decompositions. Increasing total capacity from 6 to 16 reduces mean waiting in every regime, consistent with high-utilization queueing predictions. Raising the KPI health-to-cost weight ratio increases clinical effort while weakening budget sustainability. The bonus-sharpness sweep is non-monotone because the local derivative $B^{\text{pool}}\kappa s_j(1-s_j)$ is weak when the tournament is flat and also weak when shares saturate.

Figure 9 isolates the bonus-pool channel behind the main-text Goodhart result. Larger pools reduce final funds in all three regimes. In the balanced regime, KPI–true-health correlation falls from -0.447 at $B^{\text{pool}} = 0$ to -0.839 at $B^{\text{pool}} = 15$, showing that stronger measured incentives can amplify a misaligned proxy rather than repair it.

G L2 Additional Diagnostic: Flexible Capacity as a Coordination Test

The main-text L2 ablation turns on KPI capacity steering and shows that adding flexible capacity can raise waiting. This appendix repeats the flex-pool sweep with capacity steering turned off ($\xi = 0$, `kpi_steering_mode=none`, so the flexible subpool is allocated by the static base-capacity shares rather than by last-period KPI scores.

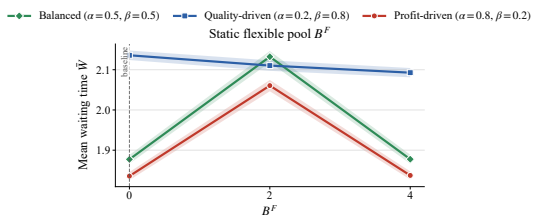


Figure 10: L2 steering-off flexible-pool diagnostic. Mean waiting (30 seeds, $T = 200$) as a function of the flexible-pool size B^{flex}, holding routing, team specialization, and total capacity fixed while disabling KPI capacity steering. Three regimes are shown; the dashed line marks the L2 baseline.

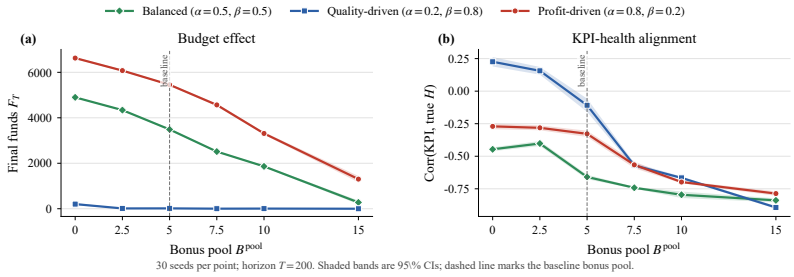


Figure 9: L2 bonus-pool ablation. Points are 30-seed means over horizon $T = 200$; shaded bands are 95% confidence intervals and the dashed line marks the baseline pool. Larger pools reduce funds and can weaken KPI-health alignment.

Figure 10 sweeps $B^{\text{flex}} \in \{0, 2, 4\}$ under the three representative regimes of §5.1. With steering off, the adverse main-text slope disappears: mean waiting is essentially unchanged from $B^{\text{flex}} = 0$ to 4 in the balanced regime ($1.88 \rightarrow 1.88$) and the profit-driven regime ($1.84 \rightarrow 1.84$), and falls modestly in the quality-driven regime ($2.14 \rightarrow 2.09$). The intermediate $B^{\text{flex}} = 2$ point rises in the balanced and profit-driven regimes, but we interpret this as a finite-team integer-allocation effect: total capacity is held fixed, so increasing B^{flex} capacity between dedicated and flexible slots, and a small flexible subpool can temporarily worsen queue–team mismatch before a larger subpool covers more teams. Therefore, flexible capacity by itself does not reproduce the adverse waiting increase seen in the main L2 panel; the increase comes from the KPI-steering rule that consumes the flexible subpool.

When capacity steering is active, the flex pool is reallocated toward teams with high KPI-steering scores, which in these rollouts are teams already adjusting triage and effort to chase bonus share. When steering is disabled, the flexible subpool is no longer systematically allocated to those teams, so it stops reinforcing the bonus-driven behavior that raised waiting in the main L2 ablation. The result shows that flexible capacity is not automatically beneficial: its effect depends on the allocation rule. In this configuration, B^{flex} helps only when the rule assigning it addresses queue imbalance rather than amplifying the coordination problem created by the KPI tournament.

H L1/L2 Experimental Setup

Table 3 centralizes the run settings used by the L1 phase diagram and the L2 administrative-lever ablations. These settings are reported here for reproducibility; the main text uses only the mechanism-level findings.

I External Stylized-Fact Validation

Table 4 summarizes the external stylized facts used to validate MEDI-SIM's provider-response dynamics. The table is not intended as a quantitative calibration exercise. It checks whether simulator interventions move the same provider-response channels in the same qualitative direction as the healthcare incentive and operations literature predicts.

J Strategic Policy-as-Code: Search Diagnostics

This part provides four pieces of structural evidence behind the L3 results of §5.3: validation search curves, a warm-start ablation that pinpoints the role of the curated library, method comparisons under each social objective that contextualize the headline numbers in Table 1, and a $K = 300$ diagnostic that characterizes the shape of the search trajectory.

The main L3 runs use ChatGPT-5.4 as the code-mutation model with $K = 200$ iterations, 3 islands, 30 individuals per island, LLM temperature 0.4, migration size 2, evolution seeds $\{101, 202, 303\}$, validation seeds $\{404, 505, 606, 707, 808\}$, and held-out test seeds $\{909, 1001, 1103, 1207, 1301\}$. The warm-start library and its diversity ablations are described in §J.1.

Figure 11 records the validation dynamics for the $K = 200$ runs. Welfare and profit objectives show short early improvements followed by plateaus, while the mixed objective reaches its best validation candidate quickly and then mostly explores lower-fitness variants. We use

Layer	Scope	Values	Notes
L1/L2	Shared rollout protocol	Horizon $T = 200$; seeds $\{1, \ldots, 30\}$; reported values are seed means unless a confidence interval is shown.	Both layers use the native hospital administrator and provider-response rules of §3; no AlphaEvolve policy is injected.
L1	Incentive phase diagram	$\alpha, \beta \in \{0.0, 0.1, \ldots, 1.0\}$, giving an 11×11 grid.	The sweep varies only provider financial sensitivity α and quality sensitivity β; all other administrative settings stay fixed.
L1	Default administrative and environment settings	$B^{\text{tot}} = 10$, $B^{\text{flex}} = 0$, $B^{\text{pool}} = 5.0$, $\kappa = 2.0$, $(w_H, w_W, w_{\text{rej}}, w_C) = (1.0, 0.5, 0.5, 0.3)$, $\xi = 0$.	Strategic routing is disabled; `kpi_steering_mode=none`; probabilistic up-coding, KPI-gaming tilt, and KPI-weighted effort feedback are enabled.
L2	Representative regimes	Profit-driven $(\alpha, \beta) = (0.8, 0.2)$; quality-driven $(0.2, 0.8)$; balanced $(0.5, 0.5)$.	Each L2 axis is swept one at a time around these three L1 anchor regimes.
L2	Main one-at-a-time lever sweeps	$q \in \{0, 0.05, 0.10, 0.20, 0.35, 0.50\}$; $B^{\text{pool}} \in \{0, 2.5, 5, 7.5, 10, 15\}$; $\kappa \in \{0, 1, 2, 3, 4, 5\}$; $B^{\text{tot}} \in \{6, 8, 10, 12, 14, 16\}$; $w_H/w_C \in \{0.5, 1, 2, 3.333, 4, 5\}$; $B^{\text{flex}} \in \{0, 2, 4\}$.	The audit sweep sets `audit_base_prob`$= q$ and `audit_slope`$= 0$; the w_H/w_C sweep holds $w_H + w_C$ fixed at the L1 baseline.
L2	Main baseline and steering settings	$B^{\text{tot}} = 10$, $B^{\text{flex}} = 0$, $B^{\text{pool}} = 5.0$, $\kappa = 2.0$, $(w_H, w_W, w_{\text{rej}}, w_C) = (1.0, 0.5, 0.5, 0.3)$, audit baseline $q = 0.10$, $\xi = 1.0$.	Strategic routing is disabled; `kpi_steering_mode=none`; probabilistic up-coding, KPI-targeting tilt, and KPI-weighted effort feedback are enabled.
L2	Steering-off flexible-capacity diagnostic	$B^{\text{flex}} \in \{0, 2, 4\}$ under the same three regimes, with $B^{\text{tot}} = 10$, $\xi = 0$, and `kpi_steering_mode=none`.	This diagnostic isolates whether the adverse waiting slope comes from flexible capacity itself or from the KPI-based allocation rule.

Table 3: L1/L2 experimental setup and hyperparameters. B^{tot} is total capacity, B^{flex} is the flexible-capacity pool, and B^{pool} is the KPI bonus pool.

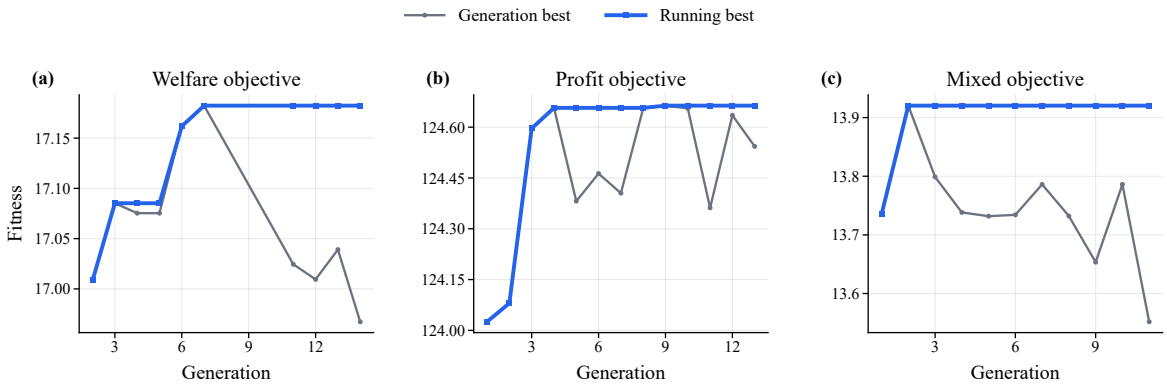


Figure 11: L3 AlphaEvolve validation search curves. Each panel shows generation-best and running-best fitness under a different evaluator with K=200. These curves are diagnostic: the main-text L3 claim rests on held-out rollout profiles in Table 1, while the curves show how each objective shapes the search trajectory.

these curves as search-process diagnostics; the policy comparison itself is based on the held-out rollout profiles in Table 1.

J.1 Warm-start library and the diversity ablation

The L3 warm-start library contains nine policies: a *neutral* template (constants equal to balanced defaults), a *fixed* heuristic that mirrors the L1 baseline, a *greedy-profit* policy with high α and aggressive coding, a *greedy-quality* policy with high β and conservative coding, three access-oriented variants (high-α acceptance, high-capacity, high-flex), and two coding-aggression variants (aggressive and conservative). Library diversity is the only configuration parameter that meaningfully changes the mixed-objective outcome at $K = 200$.

The ablation is reported as three nested libraries. With the *neutral-only* library, selection on validation fitness returns the seed itself at 13.545 and the best evolved candidate

reaches only 13.351; search cannot improve over its starting point. After removing the profit-side seeds, which the library still contains the welfare-leaning policies but no aggressive-coding warm start, the process produces the best evolved candidate at 13.401. The *full* nine-policy library produces a selected aggressive-coding warm start at 13.607 and then refines to 13.876 on held-out seeds, the value reported in Table 1. The reading we take from this nested set is clear: the gain from search over the best warm start (+0.27 fitness) is comparable to the gain from library expansion alone (+0.21), and search produces no improvement above the seed when the seed is not diverse. We therefore present AlphaEvolve over Medi-Sim as a feasibility demonstration of program search over the policy class of §4 and do not claim that current search procedures can rediscover the mixed family from scratch.

J.2 Method comparison under each social objective

Figure 12 is the main supporting figure for the mixed-objective L3 result. The discounted-return panel shows AlphaEvolve essentially tied with the profit baseline; the doctor-utility, mean-effort, and doctor-margin panels show that this tie is achieved with a substantially different operational profile: lower mean effort, higher doctor utility, and higher doctor margin than Fixed. The two bonus-related panels are the most informative. Mean bonus and bonus pressure both fall by about 60% relative to the Fixed baseline, even though Fixed and AlphaEvolve produce comparable discounted returns. The searched mixed policy therefore retains the macroeconomic return while structurally lowering the local bonus pressure $B^{\text{pool}}\kappa s_j(1-s_j)$ that drives the KPI-targeting residual identified in §E.5.

The single-objective comparisons in Figures 13 and 14 sharpen the interpretive picture in two ways. Under the pure-profit objective, the Profit warm start is already close to a local optimum on Fitness; AlphaEvolve captures only a small additional margin, visible as minor gains in discounted return, doctor utility, and doctor margin, while pushing the coding-heavy risk profile slightly further. Adding the Neutral baseline shows that the return gap between Neutral and Profit is moderate, but the operational gap is much larger: Neutral uses substantially more effort, bonus, and bonus pressure while still remaining below the profit-oriented return. Under the pure-welfare objective, AlphaEvolve also improves the operational profile relative to the Quality baseline. The doctor-margin panel rises from near zero to about 1.4, indicating that the searched welfare policy reduces the excessive effort cost of the quality-driven baseline. At the same time, it preserves the access improvements rewarded by the welfare objective, rather than gaining margin by rejecting or delaying patients.

J.3 $K = 300$ comparison panels

The four-method comparisons in Figures 15–17 reproduce the same qualitative policy families at the longer search budget. The Fixed, Profit, and Quality bars are unchanged because they are evaluations of fixed warm starts. The AlphaEvolve bars move in objective-specific directions: the pure-profit and pure-welfare runs push further along their respective objectives, while the mixed run improves waiting and high-complexity deferral but does not dominate the $K = 200$ mixed policy on held-out fitness or aggregate violations. We therefore treat $K = 300$ as a diagnostic budget check rather than as a replacement for the main $K = 200$ result.

J.4 $K = 300$ diagnostic: trajectory shape

Extending the search budget from $K = 200$ to $K = 300$ does not change the held-out conclusion: the $K = 300$ mixed run lowers high-complexity deferral but does not beat the $K = 200$ policy on held-out fitness or overall violations, and we therefore keep $K = 200$ as the main mixed result. The $K = 300$ trace is nonetheless diagnostic about the *shape* of program-space search over Π_A. Figure 18 shows that improvement is concentrated in three discrete events at iterations 198, 213, and 273, separated by long intervals of zero improvement. The fitness increments are small (+0.047, +0.002, and +0.066), and only two of the three events touch the administrative side of the policy bundle. This piecewise-constant trajectory shape is a feature of the regulated DSL of §4: the program space contains many syntactically valid local perturbations whose

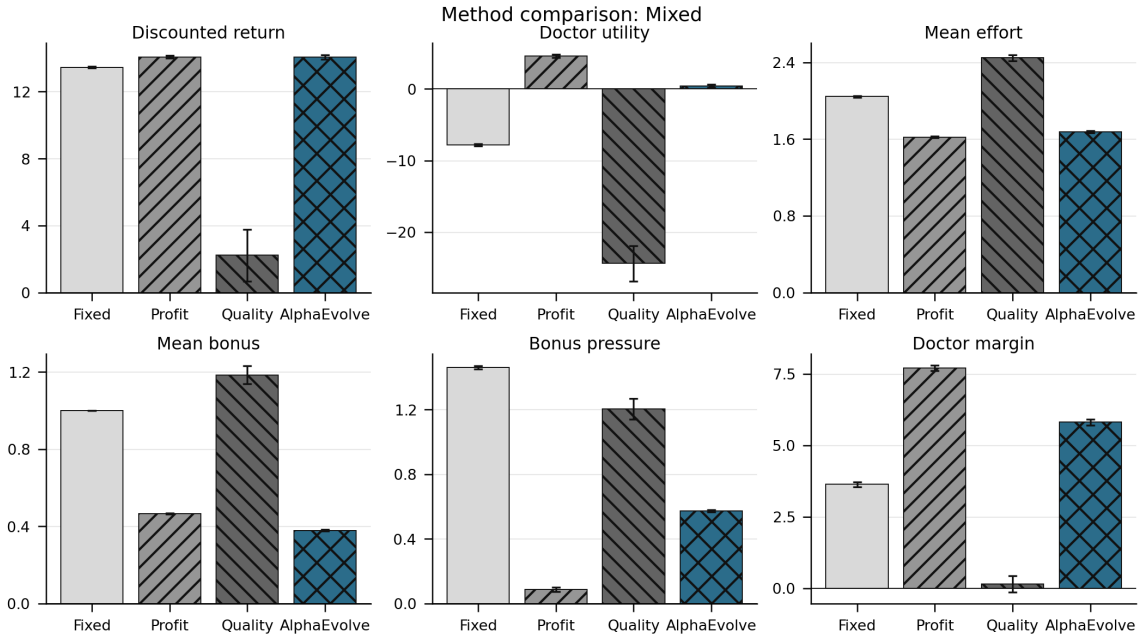


Figure 12: $K = 200$ mixed-objective four-method comparison. Bars are means over the five held-out test seeds; whiskers are 95% bootstrap intervals. The AlphaEvolve column is the central qualitative finding: it achieves profit-comparable discounted return while reducing mean bonus and bonus pressure by about 60% relative to Fixed, with doctor margin recovered to within 25% of the Profit baseline.

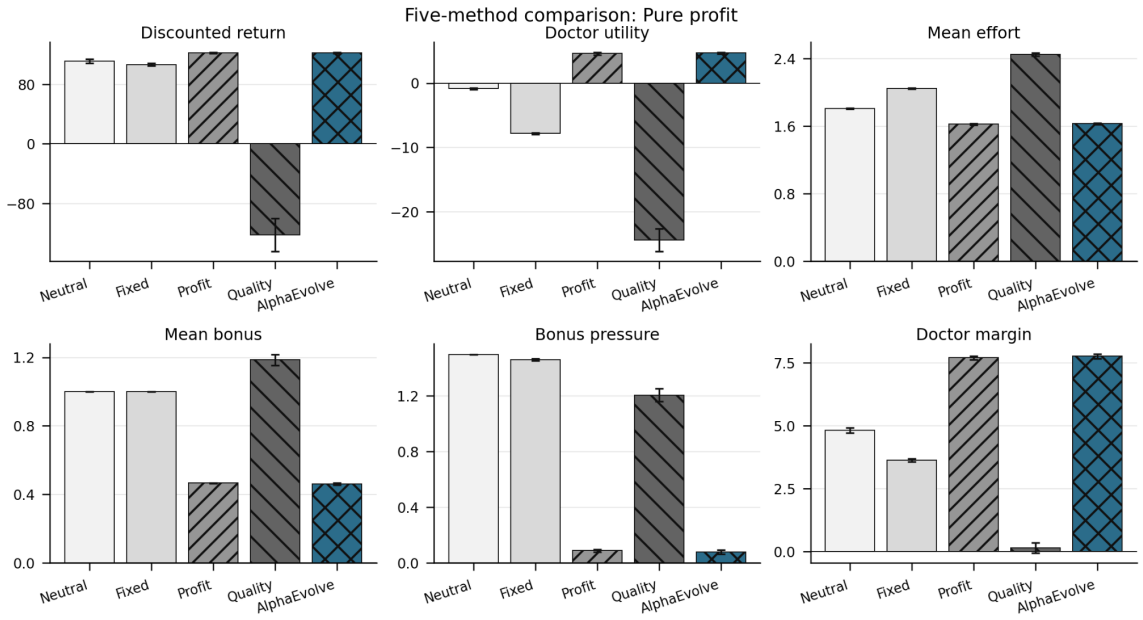


Figure 13: $K = 200$ pure-profit five-method comparison. Bars are means over the five held-out test seeds. Adding the Neutral baseline reveals that Neutral achieves a lower but still nontrivial return despite substantially higher mean effort and bonus pressure; AlphaEvolve makes a small profit-family refinement visible in return, doctor utility, and doctor margin.

simulated rollouts produce indistinguishable evaluator returns, and meaningful improvements arrive only at the rare iterations where a coordinated multi-field edit moves the mechanism across one of the regime boundaries identified in §E.

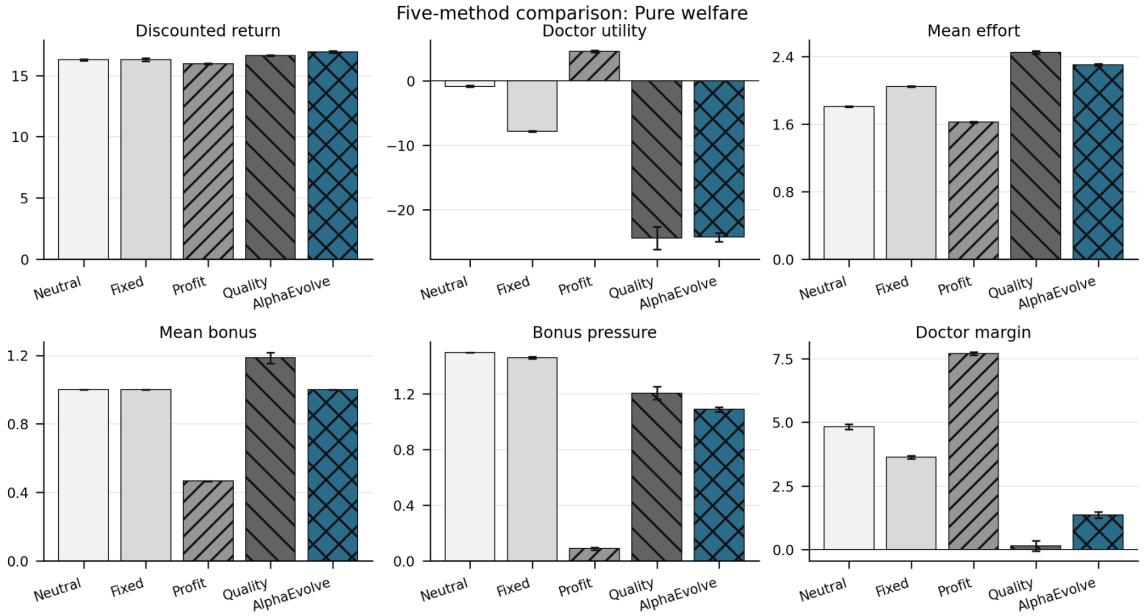


Figure 14: $K = 200$ pure-welfare five-method comparison. Bars are means over the five held-out test seeds. The AlphaEvolve refinement of the welfare family lifts discounted return above all four baselines while keeping mean effort below Quality and lifting doctor margin from near zero to $\sim$1.4, indicating that the search has trimmed the gold-plating slack identified in §E.4.

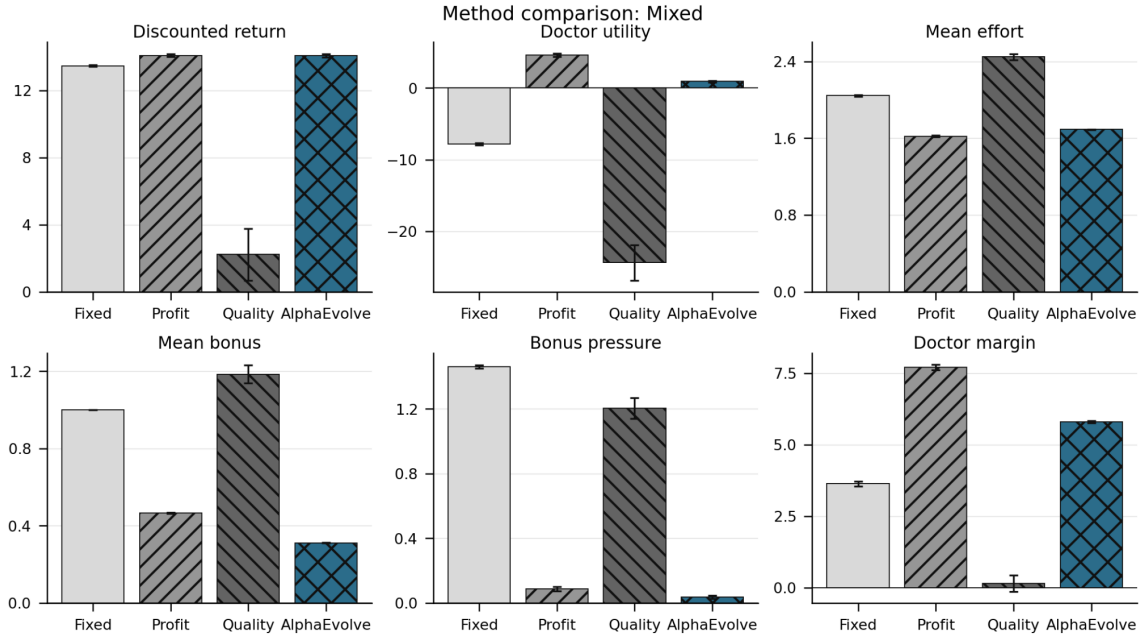


Figure 15: $K = 300$ mixed-objective four-method comparison, the longer-budget counterpart of Figure 12. Bars are means over the five held-out test seeds. The qualitative picture is unchanged from $K = 200$: AlphaEvolve achieves discounted return on par with Profit while leaving mean bonus and bonus pressure suppressed, consistent with the trajectory in Figure 18.

Table 5: $K = 300$ mixed running-best trace. Update codes summarize the policy fields changed relative to the previous running-best step. "Gap" is the strategic-delay gap.

Step	Iter.	Update	Fitness	Wait	Reject	Gap
1	0	neutral	13.735	1.797	0.022	0.328
2	198	admin + doctor	13.782	1.628	0.050	0.233
3	213	doctor only	13.784	1.606	0.050	0.236
4	273	admin + doctor	13.850	1.602	0.049	0.238

Table 5 decomposes these update events at the metric level. Each improvement event corresponds to a clear trade: rejection rises from 0.022 to $\sim$ 0.05 as the policy stops absorbing patients it cannot treat profitably, the strategic-

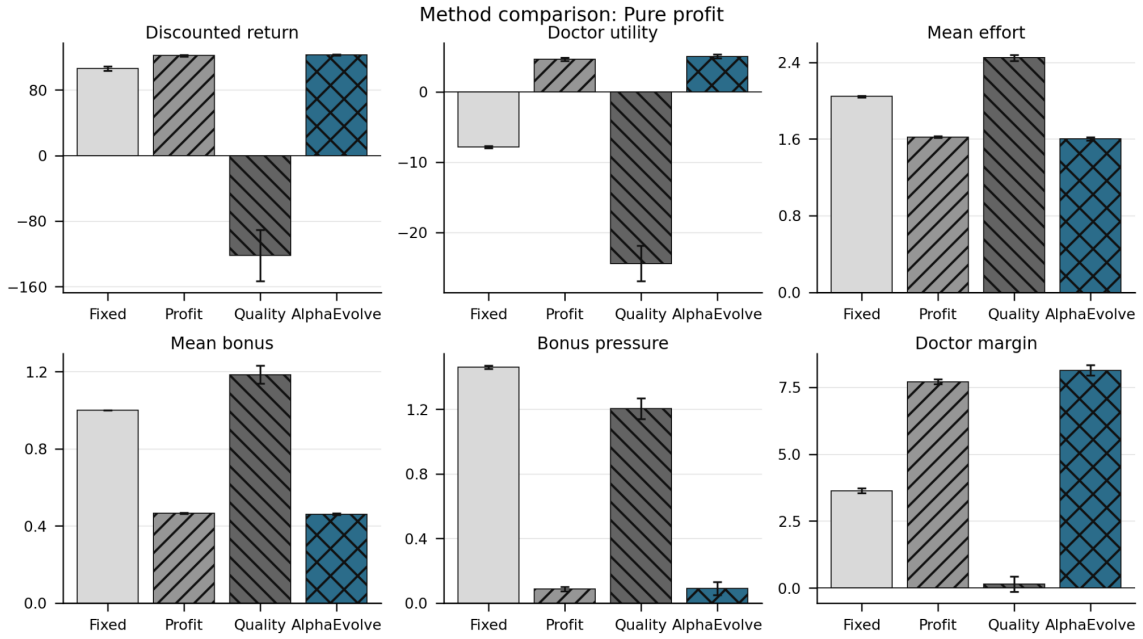


Figure 16: $K = 300$ pure-profit four-method comparison. Bars are means over the five held-out test seeds. The longer budget gives AlphaEvolve a small gain over Profit in discounted return, doctor utility, and doctor margin, while preserving the same profit-oriented risk profile.

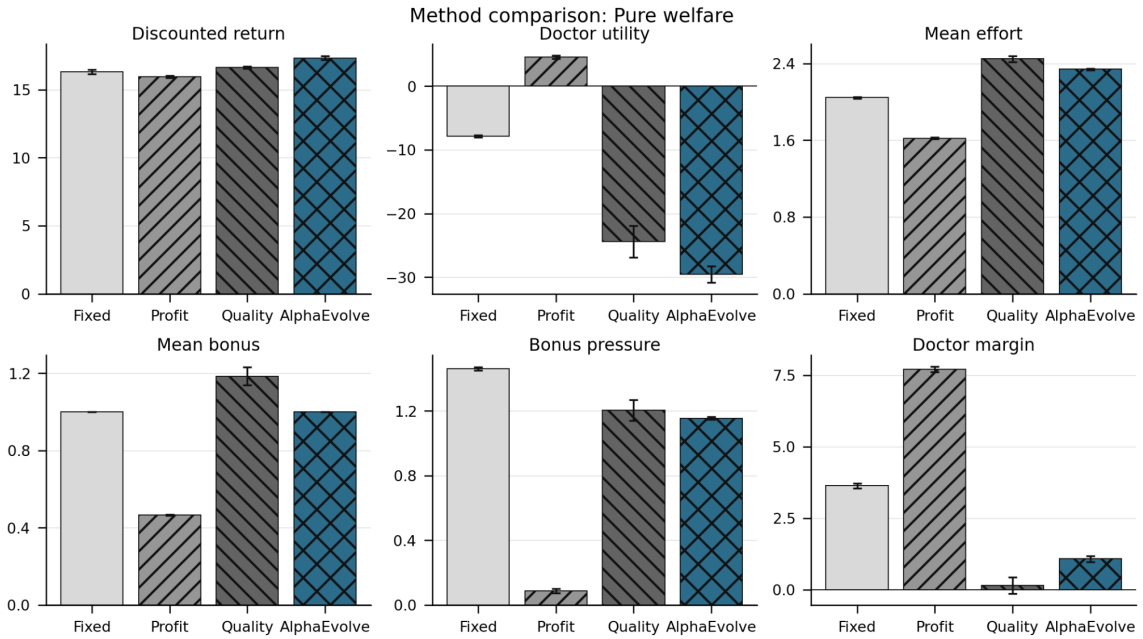


Figure 17: $K = 300$ pure-welfare four-method comparison. Bars are means over the five held-out test seeds. The welfare AlphaEvolve refinement extends the discounted-return lift seen at $K = 200$ and continues to suppress Quality-style gold-plating on the effort and bonus panels.

delay gap falls from 0.328 to ~ 0.24 as the rejection channel takes over the cost-shedding role the delay channel had carried, and mean waiting falls in lockstep. The qualitative pattern is the same channel substitution documented in §E.2, now observed inside a single search trajectory. The complete diagnostic policy sketch is given in Listing 4 below.

J.5 Discovered policy structures

The first three listings below summarize the final code structures produced by the $K = 200$

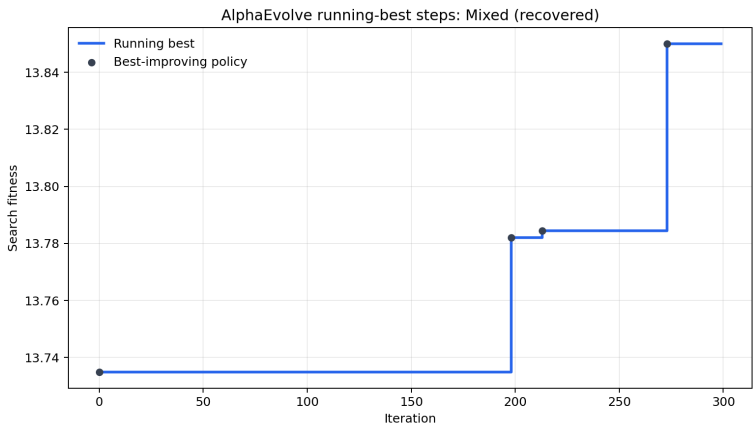


Figure 18: $K = 300$ mixed-objective running-best trace. Each step is a search iteration; the first marker is the neutral seed and later markers indicate iterations at which AlphaEvolve discovered a policy that improved the incumbent search fitness. The trajectory is piecewise constant with three update events at iterations 198, 213, and 273; Table 5 records the code-level edits associated with each event.

search under each social objective. Each listing reproduces the final field-level policy code; auxiliary dictionary syntax is omitted for readability and the full executable policies are saved by the experiment pipeline. The structures should be read against the warm-start ablation of §J.1: search refines, but does not invent, the families inherited from the library.

Welfare family (Listing 1). The welfare family pushes α down to 0.20 and β to 1.00, then adds state-conditional capacity expansion (+1 if any of queue, waiting, or utilization stress is detected) and a flex pool that grows under the same conditions. The hospital KPI weight vector tilts toward health, waiting, and rejection ($w_H = 1.80$, $w_W = 0.90$, $w_{\text{rej}} = 1.70$) while suppressing the cost weight to $w_C = 0.05$. The coding-side expression is dominated by the ethics term (-2.50), which keeps up-coding at zero across all rollouts.

Listing 1: Pure-welfare policy sketch.

```
alpha = 0.20
beta = 1.00
total_capacity = max(9, min(13, current_total_capacity
    + I((queue_total > 6.0) or (wbar > 1.4) or
      (utilization > 0.88))
    - I((queue_total < 2.0) and (utilization < 0.50))))
bonus_pool = 5.0
flex_pool = min(max(1, 3 + I((wbar > 1.2) or
      (queue_total > 5.0)
    or (queue_max > 2.0)) + I(utilization > 0.85)),
    max(1, current_total_capacity))
xi = 0.0
wH = 1.80; wW = 0.90; wrej = 1.70; wC = 0.05; kappa =
      1.7
effort = base_effort + effort_quality_pressure
triage_accept = (clinical_triage_score +
      margin_pressure + quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
      + kpi_gaming_pressure
    + 0.15 * urgency + 0.05 * wait) >= accept_threshold
triage_defer = (not ((clinical_triage_score +
      margin_pressure + quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
      + kpi_gaming_pressure
    + 0.15 * urgency + 0.05 * wait) >=
      accept_threshold)) and
    ((clinical_triage_score + margin_pressure +
      quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
    + 0.5 * kpi_gaming_pressure + 0.10 * urgency) +
      defer_pressure
    >= defer_threshold)
request_bed = bed_available and ((clinical_bed_score +
      0.20 * quality_pressure
    + bed_quality_pressure - bed_cost_pressure +
      bed_availability_signal
    - bed_fatigue_pressure + 0.05 * urgency) >=
      bed_threshold)
candidate_score = 0.20 * upcode_pressure - 1.50 *
      audit_penalty
    - 2.50 * ethics_pressure - 0.45 * coding_gap
```

Profit family (Listing 2). The profit family inverts every one of these moves. The hospital administrator commits to $\alpha \approx 0.95$ except when waiting or rejection signals a near-failure mode (in which case α falls and β rises by 0.15–0.20 as a safety reflex), keeps KPI steering active at $\xi = 1.5$, and lets a small flex pool expand under queue and rejection stress. The bonus pool scales linearly with last-period profit, and the hospital KPI weight on cost ($w_C = 1.70$) dominates all other weights. The coding-side expression doubles its weight on up-coding pressure and gain and halves its ethics-pressure coefficient: the searched profit policy actively manages coding aggressiveness as a continuous variable.

Listing 2: Pure-profit policy sketch.

```
alpha = clip(0.95 - 0.15 * I((wbar > 4.0) or (rbar >
      0.15)), 0.0, 1.0)
beta = clip(0.05 + 0.20 * I((wbar > 4.0) or (rbar >
      0.15)), 0.0, 1.0)
```

```
total_capacity = max(10, min(12, current_total_capacity
    + I((((wbar > 5.0) or (rbar > 0.18) or (queue_total
     > 24.0))
        and (utilization > 0.92)) and (funds > 70.0)
        and (profit_last > 0.0))
    - I((utilization < 0.62) and (wbar < 0.5)
        and (current_total_capacity > 10))))
bonus_pool = clip(1.20 + 0.032 * max(0.0, profit_last)
    - 0.70 * I((profit_last < 0.0) or (rbar > 0.14)),
     0.25, 4.1)
flex_pool = min(2, min(max(1, current_total_capacity),
    max(0, current_flex_pool
        + I(((queue_total > 9.0) or (queue_max > 4.0))
     and (rbar > 0.05))
        - I((not (((queue_total > 9.0) or (queue_max >
     4.0))
            and (rbar > 0.05))) and (current_flex_pool
     > 0)))))
xi = 1.5
wH = 0.15; wW = 0.05; wrej = 0.15; wC = 1.70; kappa =
     3.5
effort = base_effort + effort_quality_pressure
triage_accept = (clinical_triage_score +
     margin_pressure + quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
     + kpi_gaming_pressure
    + 0.22 * urgency + 0.06 * wait) >= accept_threshold
triage_defer = (not ((clinical_triage_score +
     margin_pressure + quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
     + kpi_gaming_pressure)
    >= accept_threshold)) and ((clinical_triage_score +
     margin_pressure
    + quality_pressure + bed_availability_signal -
     triage_fatigue_pressure
    + kpi_gaming_pressure) + defer_pressure >=
     defer_threshold)
request_bed = bed_available and ((clinical_bed_score +
     bed_quality_pressure
    - bed_cost_pressure + bed_availability_signal -
     bed_fatigue_pressure)
    >= bed_threshold)
candidate_score = 0.22 * upcode_gain + 0.78 *
     upcode_pressure
    - 1.14 * audit_penalty - 1.16 * ethics_pressure -
     0.26 * coding_gap
```

Mixed family (Listing 3). The calibrated mixed family is the cleanest qualitative result of the L3 layer and the only policy in our search that simultaneously achieves three properties: profit-comparable discounted return, zero up-coding on held-out rollouts, and a halving of rejection relative to the profit baseline. Its structure is informative. The hospital administrator fixes $\alpha = 0.5$, adapts β upward only when access stress is detected, and leaves total capacity, the flex pool, and KPI steering unchanged; the bonus pool grows with both profit and access stress but contracts sharply under insolvency; hospital KPI weights are balanced with a primary tilt toward health and rejection. The candidate-coding score is the structural innovation: it carries a moderate positive weight on `upcode_pressure` as a continuous lever, balanced against an explicit -100 indicator penalty that snaps to a regime that strictly forbids coding deviations above 0.20. This indicator term is what drives up-coding to exactly zero on held-out seeds while leaving the gradient signal usable in early-training rollouts—a hand-engineered version of the same hard-threshold-plus-smooth-shaping idiom that has emerged in several published reward-design studies.

Listing 3: $K = 200$ calibrated mixed policy sketch.

```
alpha = 0.5
beta = clip(0.35 + 0.20 * I((wbar > 4.0) or (rbar >
     0.15)), 0.0, 1.0)
bonus_pool = clip(1.45 + 0.012 * max(0.0, profit_last)
    + 0.25 * I((wbar > 1.8) or (rbar > 0.06))
    - 0.55 * I(profit_last < 0.0), 0.75, 4.25)
total_capacity = current_total_capacity
flex_pool = current_flex_pool
xi = 0.0
wH = 1.0; wW = 0.20; wrej = 1.10; wC = 0.10; kappa = 2.0
effort = base_effort + effort_quality_pressure
triage_accept = (clinical_triage_score +
     margin_pressure + quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
    + kpi_gaming_pressure) >= accept_threshold
triage_defer = (not ((clinical_triage_score +
     margin_pressure + quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
    + kpi_gaming_pressure) >= accept_threshold)) and
     ((clinical_triage_score
    + margin_pressure + quality_pressure +
     bed_availability_signal
    - triage_fatigue_pressure + kpi_gaming_pressure) +
     defer_pressure
    >= defer_threshold)
request_bed = bed_available and ((clinical_bed_score +
     bed_quality_pressure
    - bed_cost_pressure + bed_availability_signal -
     bed_fatigue_pressure)
    >= bed_threshold)
candidate_score = 0.85 * upcode_pressure + 0.08 *
     upcode_gain
    - 1.00 * audit_penalty - 0.85 * ethics_pressure
    - 0.15 * coding_gap - 100.0 * I(coding_gap > 0.20)
```

$K = 300$ diagnostic trace (Listing 4). The $K = 300$ trace produces a closely related but distinct mixed family. Listing 4 gives the neutral seed and then records only the code-level deltas at each running-best update in Table 5; the complete executable policies are stored in the experiment artifacts. The main edits occur in two stages: iteration 198 moves the administrative policy into an access-stress rule with active KPI steering, and iterations 213 and 273 refine the bonus, rejection-weight, triage, and coding expressions. The $K = 300$ family improves waiting and high-complexity deferral, but it does not dominate the $K = 200$ family on held-out fitness or the aggregate violation score; we therefore keep $K = 200$ as the main result, while using the trace to show that both budgets converge on the same coding-gap idiom.

Listing 4: $K = 300$ mixed running-best trace deltas.

```
# Iteration 0: neutral seed.
alpha = 0.5; beta = 0.5
total_capacity = current_total_capacity; flex_pool =
     current_flex_pool
bonus_pool = 5.0; xi = 0.0
```

```
wH = 1.0; wW = 0.20; wrej = 1.0; wC = 0.10; kappa = 2.0
triage_accept = (clinical_triage_score + margin_pressure +
      quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
    + kpi_gaming_pressure) >= accept_threshold
triage_defer = (not ((clinical_triage_score + margin_pressure +
      quality_pressure
    + bed_availability_signal - triage_fatigue_pressure
    + kpi_gaming_pressure) >= accept_threshold)) and
      ((clinical_triage_score
    + margin_pressure + quality_pressure + bed_availability_signal
    - triage_fatigue_pressure + kpi_gaming_pressure) +
      defer_pressure
    >= defer_threshold)
request_bed = bed_available and ((clinical_bed_score +
      bed_quality_pressure
    - bed_cost_pressure + bed_availability_signal -
      bed_fatigue_pressure)
    >= bed_threshold)
candidate_score = upcode_pressure - audit_penalty -
      ethics_pressure

# Iteration 198: delta relative to neutral seed.
alpha -> clip(0.65 - 0.08 * I((wbar > 4.0) or (rbar > 0.15)),
      0.0, 1.0)
beta  -> clip(0.30 + 0.25 * I((wbar > 4.0) or (rbar > 0.15)),
      0.0, 1.0)
total_capacity -> max(10, min(12, current_total_capacity
    + I((((wbar > 5.0) or (rbar > 0.18) or (queue_total > 24.0))
        and (utilization > 0.92)) and (funds > 70.0)
        and (profit_last > 0.0))
    - I((utilization < 0.62) and (wbar < 0.5)
        and (current_total_capacity > 10))))
bonus_pool -> clip(1.50 + 0.015 * max(0.0, profit_last)
    - 0.50 * I(profit_last < 0.0), 0.75, 4.25)
flex_pool -> min(2, min(max(1, current_total_capacity),
    max(0, current_flex_pool
        + I((queue_total > 10.0) and ((rbar > 0.06) or (wbar >
      2.0)))
        - I((not ((queue_total > 10.0)
            and ((rbar > 0.06) or (wbar > 2.0))))
            and (current_flex_pool > 0)))))
xi -> 1.5
wH,wW,wrej,wC,kappa -> 0.85, 0.05, 1.05, 0.80, 2.2
triage_accept: add 0.14 * urgency + 0.04 * wait
triage_defer: add 0.14 * urgency + 0.04 * wait to accept gate;
              add 0.08 * urgency + 0.03 * wait to defer score
request_bed: add 0.08 * urgency + 0.04 * wait
candidate_score -> 0.85 * upcode_pressure + 0.08 * upcode_gain
    - 1.00 * audit_penalty - 0.85 * ethics_pressure
    - 0.15 * coding_gap - 100.0 * I(coding_gap > 0.20)

# Iteration 213: delta relative to iteration 198.
triage_accept: urgency coefficient 0.14 -> 0.16
triage_defer: accept-gate urgency 0.14 -> 0.16;
              defer-score urgency 0.08 -> 0.09
candidate_score -> 0.80 * upcode_pressure + 0.10 * upcode_gain
    - 1.10 * audit_penalty - 0.95 * ethics_pressure
    - 0.25 * coding_gap - 100.0 * I(coding_gap > 0.20)

# Iteration 273: delta relative to iteration 213.
bonus_pool -> clip(1.35 + 0.012 * max(0.0, profit_last)
    - 0.60 * I((profit_last < 0.0) or (rbar > 0.08)), 0.75, 3.75)
wrej -> 1.15
triage_accept: urgency 0.16 -> 0.22; wait 0.04 -> 0.05;
               add 0.03 * complexity
candidate_score -> 0.70 * upcode_pressure + 0.12 * upcode_gain
    - 1.15 * audit_penalty - 1.00 * ethics_pressure
    - 0.35 * coding_gap - 100.0 * I(coding_gap > 0.20)
```

K Implementation Details and DSL Guardrails

L1 and L2 use the native hospital administrator and provider rules of §3; no AlphaEvolve policy is injected into those layers. The main L2 ablation sweeps audit probability, bonus pool, bonus sharpness, total capacity, the health-to-cost hospital KPI weight ratio, and flexible capacity; the static-flex diagnostic of §G isolates the flexible pool under fixed routing and specialization. L3 selects candidates on validation seeds and reports held-out test results on disjoint seed splits. Unlike L1/L2, L3 may edit selected doctor-side expression constants, but only through the typed fields in Listing 5; host-side dynamics, audit application, and metric computation remain fixed. The mixed-objective scalarizer is the safety-penalized fitness of §4 with $\lambda_{\text{unsafe}} = 0.06$ and $\lambda_{\text{var}} = 0.25$, with explicit per-step penalties for high-urgency rejection, insolvency, unsafe upcoding, excessive waiting, and high-complexity deferral.

DSL guardrails. The released L3 prompt files use a small replayable interface. Listing 5 summarizes the shared guardrails. The full artifacts include the system prompt, the full-rewrite template, and the diff template; the `medisim_dsl_v1` tag identifies the DSL schema and is not itself a simulator mechanism. Candidates can edit only fixed policy expression fields and can read only exposed DSL features plus safe scalar helper calls. Simulator dynamics, patient generation, metric computation, host-side clipping, and feasibility projection are held fixed across candidates, which is what closes the audit-evasion gap discussed under desideratum (ii) of §4. For coding, the host computes `audit_penalty` from $p_{\text{audit}}(\Delta\text{CMI})$ and the fixed penalty schedule before the candidate expression is evaluated; `candidate_score` may change how strongly the coder responds to that feature, but cannot change the audit probability, penalty application, or reported metrics.

Listing 5: L3 DSL guardrail summary.

```
One executable/testable assignment-only medisim_dsl_v1
module, or exact SEARCH/REPLACE blocks.
Top-level: kind, version, admin_policy, doctor_policy.
Admin expr: alpha, beta, total_capacity, bonus_pool,
flex_pool, xi, wH, wW, wrej, wC, kappa.
Doctor expr: effort, triage_accept, triage_defer,
request_bed, candidate_score.
Calls: I, clip, min, max, abs, round, soft_gt, soft_lt.
No new state/schema fields/mechanisms/hidden calls.
Host clips/projects outputs; dynamics and metrics fixed.
```

L Terminology and Arrival-Process Note

Glossary. The main text keeps the terminology short; this appendix gives the operational meanings used in the simulator.

Hospital DRG.

Diagnosis-related group: a billable case category used in prospective hospital payment. In Medi-Sim, each patient has a true clinical group and a reported group used for settlement.

Hospital DRG-style arrivals.
Hospital patient arrivals with clinical type, urgency, tolerance, and reimbursement-relevant case weight. The phrase distinguishes the stream from identical jobs in a generic queue.

Hospital CMI.
Case-mix index. We use normalized hospital CMI as the patient's clinical complexity and as the payment-relevant weight that coding can distort.

Hospital administrator.
The leader in the Stackelberg formulation. This actor sets hospital-level levers such as incentives, audit intensity, capacity, bonus pools, KPI weights, routing, and steering.

Hospital Stackelberg game.
A leader-follower model in which the hospital administrator commits to a mechanism and hospital providers respond strategically to it.

Hospital coder and hospital coding policy.
A hospital coder maps clinical evidence to a reported billable group. The hospital coding policy is the fixed or searched rule that scores candidate reported groups before the configured coding choice rule.

Hospital KPI.
A measured hospital performance score assembled from health, waiting, rejection, and cost. It affects bonuses and can diverge from true clinical value.

Hospital provider-response channels.
Up-coding changes the reported billing group; selection changes which patients are accepted; delay keeps patients waiting when serving them is unattractive; effort changes treatment intensity; triage is the local accept/defer/reject and resource-request rule.

Hospital coding wedge and hospital measurement wedge.
The coding wedge is the gap between true clinical complexity and the reported billing group. The measurement wedge is the gap between true clinical value and the measured KPI used for bonuses or steering.

Hospital KPI targeting (Goodhart-style proxy gaming).
Provider behavior that improves a measured hospital KPI while moving away from the hospital's true clinical or access objective.

Gold-plating, skimping, and cream-skimming.
Gold-plating is excessive treatment intensity, skimping is under-provision of care effort or resources, and cream-skimming is accepting easier or more profitable patients disproportionately.

Hospital Identify–Produce–Settle (IPS).
The simulator order: identify patients and billing groups, produce care under capacity constraints, then settle reimbursement, KPI scores, and bonuses.

Hospital policy DSL.
The restricted domain-specific language used to write candidate hospital policies over approved administrative levers and provider-response expressions.

Flexible hospital capacity pool and KPI steering.
The flexible capacity pool is capacity that can be reallocated across hospital care teams. KPI steering assigns that capacity or routing priority using measured hospital performance scores.

Arrival-process scope note. The experiments use the homogeneous Poisson law in Eq. (10) because it is transparent and reproducible. The simulator does not rely on Poisson arrivals for the channel definitions. Let A_t denote the full arrival batch at period t, including patient count and attributes, and let h_{t-1} be the realized pre-period history. For any fixed arrival kernel $\mathcal{K}_t(dA_t \mid h_{t-1})$ sampled before provider actions, the one-step rollout law under policy π factors as

$$\Pr_\pi(dA_t, dY_t, dX_{t+1} \mid h_{t-1}) = \mathcal{K}_t(dA_t \mid h_{t-1}) \Pr_\pi(dY_t, dX_{t+1} \mid h_{t-1}, A_t), \tag{17}$$

where Y_t collects the identify, routing, triage, treatment, and settlement decisions. The second factor is the strategic hospital-response part studied in the paper. Replacing the Poisson kernel with a nonhomogeneous Poisson process, an over-dispersed count model, correlated service-line arrivals, seasonal mixtures, or an empirical bootstrap stream changes the distribution of states the policy sees, but it leaves the IPS order and the definitions of coding, selection, delay, effort, and KPI targeting intact, provided the same arrival kernel is used for the policies being compared.

This is the scope condition. If patient demand responds directly to the hospital policy or to reputation generated by earlier policies, the arrival kernel is no longer fixed. That feedback can be modeled as an additional demand-response channel, but it is outside the present L1/L2/L3 experiments.

External stylized fact	Anchor	Simulation intervention	Expected qualitative signature	Medi-Sim observed signature	Match strength and caveat
Diagnosis-based payment creates coding rent	Dafny (2005); Kronick and Welch (2014)	Increase profit sensitivity or payment spread; high-α/low-β region	Reported complexity or up-coding rises; funds rise; true clinical need need not rise equally	Profit-driven representative regime: up-coding = 0.226; funds = 5451.7	Strong directional match. We claim qualitative coding-rent reproduction, not real-system magnitude calibration.
Risk-adjusted reimbursement raises reported risk or coded complexity	Kronick and Welch (2014); Geruso and Layton (2020)	Strengthen reimbursement value of the coded group while the coding wedge is active	Coded risk rises relative to true patient complexity	High-profit cells show elevated up-coding; the main tables report the up-coding rate rather than coded-minus-true CMI decomposition	Partial match. Stronger if future reports add explicit coded-CMI inflation.
Profit or ownership-like incentives shift case mix	Silverman and Skinner (2004); Ellis (1998); Ma (1994)	Move toward high α at low β; compare profit-driven cells with quality and balanced cells	More profitable or easier patients are favored; high-cost/high-CMI patients face worse access	Profit-driven representative regime: high-CMI rejection gap = 0.182 alongside elevated coding and funds	Strong directional match. In L1/L2, active hospital routing is disabled, so the selection signature comes from provider triage.
Audit suppresses coding but can shift pressure elsewhere	Kuhn and Siciliani (2008)	Increase audit probability q from 0 to 0.5	Up-coding falls; selection or delay pressure may rise when the billing channel closes	Profit-driven up-coding 0.851 $\rightarrow$ 0.003; balanced up-coding 0.636 $\rightarrow$ 0.001; balanced cherry-picking 0.100 $\rightarrow$ 0.233	Strong match. The result validates channel substitution, not a calibrated audit schedule.
Measured performance targets induce gaming	Bevan and Hood (2006); Propper et al. (2010); Campbell (1979); Manheim and Garrabrant (2018)	Increase KPI salience through balanced-interior incentives, bonus pressure, or KPI steering	Measured score becomes behaviorally salient while true-health alignment can weaken	Balanced interior: lagging teams tilt accepted case mix toward easier cases (tilt = 0.341); corr(KPI, health) = -0.659	Strong match. The simulator does not reproduce a specific NHS target; it reproduces proxy-objective decoupling.
Stronger pay-for-performance can worsen true alignment when the proxy is misaligned	Holmstrom and Milgrom (1991); Baker (1992); Eijkenaar et al. (2013); Van Herck et al. (2010)	Increase the bonus pool B^{pool}	The measured proxy becomes more consequential; KPI–health alignment may worsen	Endpoint comparison: balanced corr(KPI, health) is -0.447 at $B^{\text{pool}} = 0$ and -0.839 at $B^{\text{pool}} = 15$; the smallest positive pool gives a small non-monotone uptick	Strong match. The caveat is conditional: stronger incentives are harmful here because the measured proxy is misaligned.
Quality-oriented incentives suppress coding/selection but raise effort and budget pressure	Ma (1994); Ellis (1998); Eggleston (2005)	Low α, high β regime	Lower up-coding and less cream-skimming; higher effort and weaker solvency	Quality-driven regime: up-coding = 0.027; rejection gap = -0.051; effort = 2.435; funds = 15.2	Strong match. This is a multi-tasking trade-off in the simulator's simplified clinical-production function.
Total capacity expansion reduces waiting under queueing pressure	Green (2002, 2006)	Increase total capacity from 6 to 16	Mean waiting should fall across regimes	Mean waiting falls in all three L2 regimes: profit 2.313 $\rightarrow$ 1.735, quality 2.325 $\rightarrow$ 1.896, balanced 2.303 $\rightarrow$ 1.720	Strong operations sanity check. This row validates the queueing layer rather than a strategic distortion channel.
Flexible capacity helps only under an appropriate allocation rule	Bekker et al. (2017) and hospital-flow allocation logic	Compare KPI-steered flexible pool with steering-off flexible pool	Flex capacity may fail if allocated by the wrong proxy; the adverse effect should disappear when steering no longer follows the proxy	With KPI steering, balanced waiting rises 1.88 $\rightarrow$ 2.23 as B^{flex} grows 0 $\rightarrow$ 4. With steering off, balanced and profit-driven waiting remain 1.88 $\rightarrow$ 1.88 and 1.84 $\rightarrow$ 1.84; quality falls 2.14 $\rightarrow$ 2.09	Strong special finding. Total capacity is fixed here; the experiment tests the allocation rule for the flexible subpool.

Table 4: Full external stylized-fact validation map. Rows report qualitative matches between external healthcare incentive/operations facts and Medi-Sim signatures. Match strength refers to directional reproduction under fixed simulator rules, not quantitative calibration to any real hospital system.