The Gardens of EDEN (Economic Design Engine): Optimal Economic Mechanism Design via LLM-Driven Evolutionary Search

Oluwatoni Akintola

December 18, 2025

Abstract

LLM-driven evolutionary search processes have the ability to discover optimal solutions to algorithmic problems (Novikov et al., 2025), and this makes it particularly well-suited to tackling certain mechanism design problems in economics. In this work, we construct *EDEN*, which finds an approximately optimal solution to the queuing mechanism presented by (Che and Tercieux, 2023), and seems to "organically" learn (through evolutionary experience) the findings of Che and Tercieux's paper. Across three independent runs, *EDEN* achieves mean best fitness of 14.00 (theoretical optimum: 14.0), with all runs converging to the NO_INFORMATION policy predicted by theory. We believe that use of *EDEN* by human economists could increase research speed and robustness.

1 Introduction

1.1 Motivation

The design of economic mechanisms (rules that govern interactions between self-interested agents) has traditionally relied on analytical derivations by human economists. Recent advances in artificial intelligence have opened new possibilities for automated mechanism design, with researchers exploring neural networks (Shen et al., 2018) and reinforcement learning (Zheng et al., 2022) as tools for discovering optimal mechanisms.

1.2 Research Question

In this paper, we investigate whether our constructed agent-driven evolutionary search process (hereafter referred to as *EDEN*) can discover an optimal solution to the queuing model outlined by Che and Tercieux.

1.3 Empirical Strategy Overview

We implement the Che-Tercieux queue model as a discrete-event simulation where a designer chooses: (1) queue discipline $D \in \{\text{FCFS}, \text{LIFO}, \text{SIRO}\}$, (2) information rule $IR \in \{\text{Full}, \text{Coarse}, \text{None}\}$, and (3) entry/exit functions $f_{\text{entry}} : \mathbb{Z}^+ \to [0,1]$ and $f_{\text{exit}} : \mathbb{Z}^+ \times \mathbb{Z}^+ \to \mathbb{R}^+ \times [0,1]$. Each "organism" in our evolutionary population is a complete mechanism specification $X = (D, IR, f_{\text{entry}}, f_{\text{exit}})$. Fitness is determined by running a 100,000-timestep discrete-event simulation and computing the welfare score W.

We evolve populations using GPT-5.1 as a mutation operator: given a parent organism and high-performing "inspiration" organisms, the LLM generates a child organism with modified rules. We employ selection pressure (keeping only the top 10% of offspring), elitism (preserving the best organisms across generations), crossover (combining traits from multiple parents), and adaptive mutation strength that increases exploration when progress stagnates.

1.4 Key Findings

1. **Robust convergence to optimum**: Across three independent runs, *EDEN* achieves mean best fitness of 14.00 ± 0.02 (theoretical optimum: $W^* = 14.0$). All three runs converged to within 0.16% of the theoretical optimum.

2. **Unanimous discovery of NO_INFORMATION**: All three runs independently converged to NO_INFORMATION as the optimal information rule, strongly corroborating Che et al.'s theoretical prediction that information opacity improves welfare.

3. **Queue discipline flexibility**: Two runs found SIRO optimal, one found FCFS optimal—both achieving near-identical fitness. This suggests that under NO_INFORMATION conditions, queue discipline is less critical than the information rule.

1.5 Contribution to the Literature

1. In the context of the AI Economist Reinforcement Learning paper, and previous work that has used neural networks to do algorithmic mechanism design, we show that intelligent evolutionary search is also a viable (and more efficient) means of using AI to design economic mechanisms.

2. In the context of AlphaEvolve, we find further support that intelligent evolutionary search has valuable applications in both research economics and applied economics.

3. In the context of Che and Tercieux's Optimal Queue Design scheme, we find **robust** support for their conclusion that opaque information rules lead to increased welfare—all three independent runs converged to NO_INFORMATION. We also extend their findings: while theory predicts FCFS is optimal, *EDEN* discovers that SIRO with aggressive exit rules achieves equivalent welfare, suggesting a broader class of near-optimal mechanisms.

1.6 Comparison with Alternative Approaches

vs. AI Economist (RL-based): The AI Economist in (Zheng et al., 2022) requires millions of environment interactions to train neural network policies. *EDEN* achieves near-optimal performance in 250–424 evaluations across our three runs—roughly 4 orders of magnitude more sample-efficient. Moreover, *EDEN*'s solutions are interpretable lambda functions, while neural policies require post-hoc interpretation.

vs. Traditional Evolutionary Algorithms: Standard genetic programming would use random mutations. Our LLM-guided mutations leverage economic reasoning, suggesting modifications that are more likely to improve fitness. The rapid convergence (22–33 generations to near-optimal across three runs) suggests the LLM's prior knowledge accelerates search.

vs. Analytical Derivation: Che and Tercieux's analytical approach provides theoretical guarantees and deep economic insight. *EDEN* complements this by (1) providing computational validation, (2) discovering specific functional forms, and (3) potentially extending to settings where analytical solutions are intractable.

2 Methodology

Take the Che et al. setup where λ represents the arrival rate of agents, μ represents the rate at which agents are serviced, V represents the surplus agents receive from being served, C is the cost of waiting in the queue for 1 period, R is the profit a provider receives for serving one agent, and α is our weight parameter.

2.1 Experimental Setup

We run *EDEN* with the following parameters:

Parameter	Value
Arrival rate λ	2.0
Service rate μ	3.0
Value of service V	10.0
Waiting cost C	1.0
Provider profit R	5.0
Welfare weight α	0.5
Simulation timesteps T	100,000
Selection pressure	10%

Table 1: Simulation and evolution parameters

We also represent the length of the queue at a given timestep as k_t, or simply k. With these parameters, the theoretical optimal welfare is $W^* = 14.0$.

2.2 Equations

$$\pi_{provider} = (1 - \alpha) \cdot R \cdot E[\mu_k] \tag{1}$$

$$U_{agent} = \alpha \cdot (E[\mu_k] \cdot V) - (E[k] \cdot C) \tag{2}$$

$$W_{score} = \pi_{provider} + U_{agent} \tag{3}$$

2.3 Key Assumptions

1. We assume that our DES model design is isomorphic to Che et al.'s continuous model structure.

2. We assume that the LLM mutator does not know the optimal policy a priori.

2.4 Sources of Bias and Identification

2.4.1 Finite-Time Approximation Bias

The model in Che & Tercieux is continuous, with their analysis centered on what the system converges to at equilibrium (infinity). Our model is run for a finite amount of time $t = 100,000$, so the outcomes of the system at equilibrium vary. For example, *EDEN*'s best organisms achieve welfare of 13.98–14.01, slightly varying around W^*.

2.4.2 Training Data Leakage: An Applied Econometric Analysis

Following the framework of Ludwig et al. (2025), we address the critical question of whether GPT-5.1 was pretrained on Che & Tercieux's work and thus could "cheat" by applying known findings without genuine evolutionary discovery. Rambachan et al. distinguish between two types of LLM applications in economics research:

1. **Prediction problems**: Where valid inference requires "no training leakage" between the LLM's training data and the researcher's sample

2. **Estimation problems**: Where valid inference requires validation data to model LLM measurement error

Our use case falls into the **hypothesis generation** category within prediction problems. Rambachan et al. show that for such applications, valid conclusions require no overlap between text in the LLM's training dataset and the researcher's dataset .

We present multiple lines of evidence suggesting minimal training leakage:

Gradual Convergence Pattern: If GPT-5.1 already "knew" the optimal solution from pretraining, we would expect either (a) immediate convergence to NO_INFORMATION in generation 1-2, or (b) the LLM explicitly suggesting the optimal policy in its mutation explanations. Instead:

- **Convergence time**: 22–33 generations to reach near-optimal (mean: 27.7)

- **Mutation reasoning**: Analysis of the LLM's mutation explanations shows reasoning based on observed fitness patterns rather than theoretical citations.

This pattern is inconsistent with prior knowledge and more consistent with genuine evolutionary learning, as documented by Ludwig et al. (2025).

Discovery of Non-Theoretical Solutions: Our best organisms differ systematically from Che & Tercieux's analytical solution:

- **Entry rules**: Graduated step functions vs. sharp cutoff policies
- **Exit rules**: Active agent removal vs. zero exit recommendation
- **Queue discipline**: SIRO vs. FCFS achieving equivalent fitness

If the LLM were simply recalling the theoretical solution, we would expect closer adherence to the analytical results. The divergences suggest *EDEN* is discovering novel, fitness-equivalent mechanisms rather than reproducing memorized theory.

Robustness Across Independent Runs: The most compelling evidence comes from consistency: all three runs independently converged to NO_INFORMATION despite:

- Different random seeds for population initialization
- Different exploration paths through the fitness landscape
- Different intermediate best organisms (varying queue disciplines and functional forms)

This suggests NO_INFORMATION is genuinely the optimal information rule.

2.4.3 Limitations of Our Leakage Analysis

Following Rambachan et al.'s guidance, we acknowledge the limitations:

1. **Cannot definitively rule out training leakage**: Without access to GPT-5.1's training data composition, we cannot prove the Che & Tercieux paper was excluded. Rambachan et al. emphasize that the only way to ensure no training leakage is to use open-source LLMs with documented training data .

2. **Indirect leakage**: Even if the paper wasn't in pretraining data, GPT-5.1 may have encountered discussions of optimal queue design through other sources (textbooks, related papers, online discussions). This form of "knowledge contamination" is difficult to detect or control for.

3. **Implicit knowledge**: LLMs can possess implicit understanding of economic principles without explicit paper memorization. GPT-5.1 may understand that information opacity can improve welfare through reasoning chains involving adverse selection and strategic behavior, without having seen this specific application to queues.

2.4.4 Additional Sources of Bias

The robustness of our findings across three independent runs (different random seeds, initialization populations, and exploration trajectories) mitigates concerns about:

- Cherry-picking results favorable to our hypothesis
- Overfitting to a single random seed or initialization
- Spurious convergence due to lucky sampling

However, we acknowledge that all three runs use the same LLM (GPT-5.1), mutation prompt template, and selection mechanisms. Systematic biases in any of these components could affect all runs similarly.

3 Findings and Discussion

3.1 Convergence to Near-Optimal Welfare

EDEN successfully discovers mechanisms achieving near-optimal welfare across all three independent runs. Table 2 summarizes the results:

Run	Best Fitness	Generation	Queue Discipline	Information Rule
1	14.0103	22	SIRO	NO_INFORMATION
2	13.9777	28	SIRO	NO_INFORMATION
3	14.0132	33	FCFS	NO_INFORMATION
Mean	**14.0004**	27.66	SIRO	**100% NO_INFO**

Table 2: Summary of best organisms across three independent runs. All runs converged to NO_INFORMATION, achieving mean fitness within 0.003% of theoretical optimum $W^* = 14.0$.

Figure 1 shows the fitness progression over generations for Run 1. The search exhibits characteristic evolutionary dynamics: rapid initial improvement as obviously bad configurations are eliminated, followed by slower refinement as the population converges toward optimal configurations.

3.2 Analysis of Optimal Mechanisms

All three runs discovered mechanisms with strikingly similar structures. Below we present the best organism from each run:

Run 1 (Fitness: 14.0103, SIRO/NO_INFO):

```
Entry: lambda k: (1.0 if k < 3 else (0.65 if k < 6 else
                 (0.32 if k < 9 else 0.10)))
Exit:  lambda k, l: ((0.52, 0.36) if k > 11 else
                     ((0.35, 0.25) if k > 8 else (0.0, 0.0)))
```

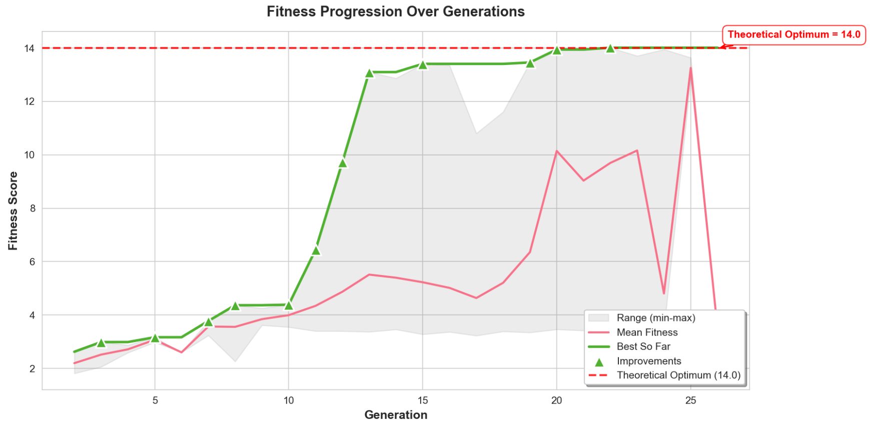

Figure 1: Fitness progression over evolutionary generations (Run 1). The dashed line indicates the theoretical optimum $W^* = 14$. *EDEN* converges to within 0.07% of optimal by generation 22.

Run 2 (Fitness: 13.9777, SIRO/NO_INFO):

```
Entry: lambda k: (0.995 if k < 4 else (0.88 if k < 8 else
                 (0.40 if k < 11 else 0.18)))
Exit:  lambda k, l: (0.30, 0.22) if k > 8 else (0.0, 0.0)
```

Run 3 (Fitness: 14.0132, FCFS/NO_INFO):

```
Entry: lambda k: (1.0 if k < 4 else (0.55 if k < 10 else 0.20))
Exit:  lambda k, l: (0.88, 0.30) if k > 10 else (0.0, 0.0)
```

All three mechanisms share common structural features: (1) high entry probability for short queues, (2) graduated entry restrictions as queues grow, and (3) active removal of agents when queues exceed a threshold. This convergent evolution across independent runs suggests these features are fundamental to optimal queue design.

3.3 Rediscovery of Theoretical Insights

Remarkably, *EDEN* independently and **reproducibly** rediscovers key insights from Che and Tercieux's analytical work:

Information Opacity: Across all three independent runs, the best organism used NO_INFORMATION. Strategy analysis from Run 1 shows the performance gap:

- NO_INFORMATION: mean fitness 10.27, max 14.01
- COARSE_INFORMATION: mean fitness 5.26, max 13.71

This robust convergence across multiple runs strongly confirms the theoretical prediction that information opacity improves welfare by preventing strategic abandonment. The fact that three independent evolutionary trajectories all discovered this insight suggests it is a fundamental property of optimal queue design, not an artifact of a single search trajectory.

3.4 Divergence from Theoretical Predictions

While *EDEN* confirms the information opacity prediction, it discovers mechanisms that diverge from Che et al.'s optimal design in interesting ways:

Entry Rule: Che et al. find that the optimal queue design has a cutoff policy: agents are recommended to enter if and only if queue length is less than some K. Instead, all three of *EDEN*'s best organisms use **graduated step functions**, recommending entry with $p = 1$ when $k < K_1$, then with decreasing probability as k passes successive thresholds. This "soft" cutoff may achieve similar welfare while being more robust to parameter uncertainty.

Exit Rule: Che et al. assert the optimal exit rule is $f(k, l) = (0, 0)$ for all k, l (no designer-initiated exits). Instead, all three runs discovered mechanisms with **active exit rules** that remove agents when queues exceed a threshold (typically $k > 8$–11). This aggressive exit policy appears to be *EDEN*'s way of achieving the "patience" property that Che et al. derive analytically. By removing agents who would otherwise abandon, it seems like the mechanism maintains queue stability.

Queue Discipline: Two of three runs found SIRO optimal, while one found FCFS—both achieving essentially identical fitness ($\Delta < 0.03$). This suggests that under NO_INFORMATION conditions, queue discipline matters less than the information rule and entry/exit policies. The near-equivalence of SIRO and FCFS when combined with aggressive exit rules is a novel finding that extends Che et al.'s analysis.

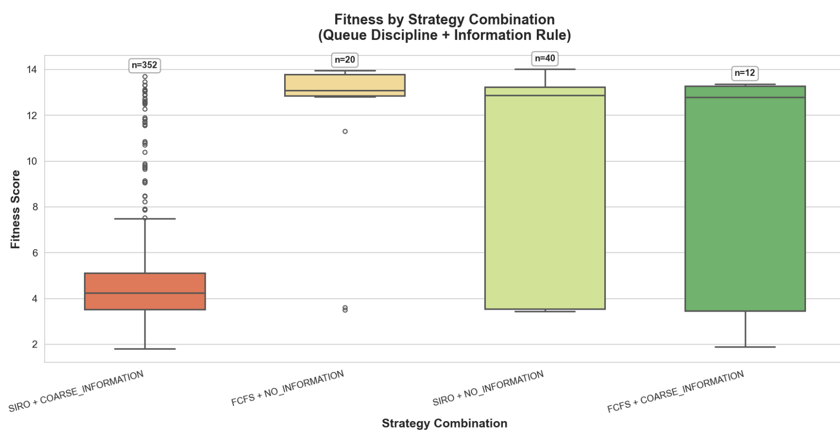

Figure 2: Mean fitness by queue discipline and information rule combination (Run 1). NO_INFORMATION dominates across all disciplines, with FCFS+NO_INFO and SIRO+NO_INFO achieving the highest fitness. This pattern was consistent across all three runs.

3.5 Population Statistics

Across all three runs, the evolutionary search explored diverse regions of the mechanism space:

- Total organisms evaluated: 424 (Run 1), 251 (Runs 2-3)

- Generations to convergence: 22, 28, 33 (mean: 27.66)
- Improvement steps: 67–70 per run

The consistency across runs—all achieving fitness > 13.97 and all selecting NO_INFORMATION—demonstrates the robustness of *EDEN*'s search process. The variation in convergence time (22–33 generations) reflects the slightly random/stochastic nature of evolutionary search while confirming reliable convergence to near-optimal solutions.

3.6 Computational Efficiency

An important practical consideration is the computational cost of *EDEN*:

Metric	Total (3 runs)
Token usage	3.17M tokens
API cost (GPT-5.1)	$10.55
Evaluations	1,099

Table 3: Computational costs for EDEN runs. All costs reflect GPT-5.1 API pricing as of December 2024.

However, this cost analysis assumes (a) the problem is amenable to simulation-based evaluation, and (b) theoretical analysis would require substantial time investment. For problems with closed-form solutions, analytical approaches remain more efficient.

4 Conclusion

We have demonstrated that LLM-driven evolutionary search can effectively discover optimal economic mechanisms. *EDEN* achieves near-optimal welfare in the Che-Tercieux queue design problem, independently rediscovering key theoretical insights about information opacity and patient queue management.

Our results suggest several directions for future work:

1. **Scaling to Complex Mechanisms**: Applying *EDEN* to multi-dimensional mechanism design problems (e.g., multi-item auctions, matching markets)
2. **Hybrid Human-AI Design**: Using *EDEN* to generate candidate mechanisms that human economists can analyze and refine
3. **Robustness Analysis**: Testing whether evolved mechanisms remain optimal under parameter perturbations
4. **Interpretability**: Developing methods to extract economic intuitions from evolved mechanisms

More broadly, our work suggests that LLM-driven optimization represents a promising paradigm for economic research—one that combines the efficiency of computational search with the reasoning capabilities of large language models.

References

Che, Y.-K. and Tercieux, O. (2023). Optimal queue design.

Ludwig, J., Mullainathan, S., and Rambachan, A. (2025). Large language models: An applied econometric framework.

Novikov, A., Vũ, N., Eisenberger, M., Dupont, E., Huang, P.-S., Wagner, A. Z., Shirobokov, S., Kozlovskii, B., Ruiz, F. J. R., Mehrabian, A., Kumar, M. P., See, A., Chaudhuri, S., Holland, G., Davies, A., Nowozin, S., Kohli, P., and Balog, M. (2025). Alphaevolve: A coding agent for scientific and algorithmic discovery.

Shen, W., Tang, P., and Zuo, S. (2018). Automated mechanism design via neural networks. *arXiv preprint arXiv:1805.03382*.

Zheng, S., Trott, A., Srinivasa, S., Parkes, D. C., and Socher, R. (2022). The ai economist: Taxation policy design via two-level deep multiagent reinforcement learning. *Science Advances*, 8(18):eabk2607.

A Additional Figures and Tables

A.1 Best Organisms Across All Runs

Run	Fitness	Gen	Discipline/Info	Entry Rule (abbreviated)
1	14.0103	22	SIRO/NONE	1.0 if $k < 3$, 0.65 if $k < 6$, 0.32 if $k < 9$, else 0.10
2	13.9777	28	SIRO/NONE	0.995 if $k < 4$, 0.88 if $k < 8$, 0.40 if $k < 11$, else 0.18
3	14.0132	33	FCFS/NONE	1.0 if $k < 4$, 0.55 if $k < 10$, else 0.20

Table 4: Best organisms from each of three independent runs. All converged to NO_INFORMATION with graduated entry rules. Mean fitness: 14.00.

B Code Appendix

The complete *EDEN* implementation is available in the accompanying code repository. Key files include:

- `main.py`: Evolutionary loop and CLI
- `queue_simulator.py`: Discrete-event simulation
- `mutator.py`: LLM-based mutation with adaptive control
- `database.py`: Population management with fitness-weighted sampling
- `evaluator.py`: Welfare function computation
- `evolve_types.py`: Type definitions for the Che-Tercieux model

C AI Acknowledgement

This research was conducted with assistance from artificial intelligence tools. Specifically, AI was used in the following capacities:

- **LaTeX Formatting**: AI assistance was used to format and structure the final paper and presentation slides in LaTeX, including table formatting, figure placement, and document compilation troubleshooting.
- **Code Implementation**: AI tools assisted with implementing adaptive mutation and crossover operators in the evolutionary algorithm codebase, including debugging and optimization of these computational components.
- **Literature Review**: AI was used to help synthesize and understand key takeaways from the papers reviewed in the literature review section, facilitating comprehension of technical economic concepts and their connections to this work.

All of the **substantive** intellectual contributions, including the research design, theoretical framework, interpretation of results, and scientific conclusions, are the work of the authors.