Simultaneous Structural and Parameter Optimization in Agent-Based Models Using Adaptive Genetic Programming

Gayani P. D. P. Senanayake, Minh Kieu, Ruggiero Lovreglio, Yang Zou, Kim Dirks, Lukas Schubotz, and Emile Chappin

***Abstract*—This study presents Adaptive Dual-OPtimization with Tree learning Genetic Programming (ADOPT-GP), a dual-loop evolutionary framework that simultaneously discovers symbolic rule structures and calibrates parameters. ADOPT-GP couples adaptive genetic programming with a two-stage parameter tuning process: rapid logistic-regression initialization followed by evolutionary calibration. Across runs, fitness improves by 20%–40% on average. Against a bilevel sequential baseline, ADOPT-GP delivers similar or better accuracy while reducing runtime by over 85%, demonstrating scalability. In a university library evacuation case, it yields diverse, interpretable rules that expose tensions between group cohesion and spatial constraints, supporting context-sensitive behaviors. The approach can advance inverse generative social science (IGSS) by linking behavioral theory with computation and offers practical tools for emergency planning.**

***Index Terms*—Agent-based modeling, dual-optimization, inverse modeling, human behavior.**

I. Introduction

AGENT-BASED models (ABMs) have emerged as a powerful approach for connecting individual behaviors to collective outcomes in social science. In mass evacuation, the collective decisions of individuals determine the overall building clearance times [1]. However, identifying underlying behavioral rules is challenging due to model uncertainty, parameter sensitivity, and stochasticity [2]. Traditional theory-first approaches struggle with this complexity, prompting a shift toward data-driven "inverse" methods.

In inverse modeling, real-world data patterns guide the discovery of behavioral rules [3]. Inverse generative social science (IGSS) automates this process by iteratively generating and testing models to fit empirical patterns, thereby discovering plausible explanations [2]. Applications range from opinion dynamics and flocking behavior [4] to residential segregation [5] and alcohol use behaviors [6]. This concept shares principles with pattern-oriented modeling (POM) [3], which similarly emphasizes matching multiple patterns at different hierarchical levels.

However, IGSS faces significant challenges: the tendency of genetic programming (GP) to overfit noisy or stochastic data [2], difficulties in jointly optimizing model structures and parameters [6], and balancing interpretability with theoretical relevance. These limitations highlight the need for methodological advances in efficient model evaluation, joint structure-parameter optimization, and improved validation methods. This research introduces Adaptive Dual-OPtimization with Tree learning Genetic Programming (ADOPT-GP), a novel methodology for simultaneous structure and parameter optimization. Key innovations include adaptive GP operators that bias crossover and mutation toward compact, high-importance subtrees, reducing bloat and stabilizing convergence to more interpretable behavioral rules. Additionally, the two-stage parameter optimization process combines rapid initial estimation with evolutionary calibration. We demonstrate this methodology using empirical data from university library evacuations.

II. Related Works

A. GP in Model Discovery

ABMs have shifted from manual hypothesis testing to automated, data-driven discovery using evolutionary methods. GP is widely used to assemble domain primitives into symbolic, interpretable rules that reproduce observed macropatterns. Gunaratne and Garibay [5] showed how GP uncovers alternative "social theories" yielding the same aggregate outcomes, while Vu et al. [6] used multiobjective GP to balance fit and complexity in alcohol-use dynamics. Related GP-based "structural calibration" spans opinion dynamics and archaeology to crowd simulations, recombining domain primitives to recover behavior. Notably, Zhong et al. [7] applied gene-expression programming to optimize crowd decision rewards, and Smith [8] evolved simple flocking rules. Together, these studies establish

Received 24 March 2025; revised 29 July 2025 and 19 September 2025; accepted 27 October 2025. This work was supported by the University of Auckland Doctoral Scholarship. *(Corresponding author: Gayani P. D. P. Senanayake.)*

Gayani P. D. P. Senanayake, Minh Kieu, Yang Zou, and Kim Dirks are with the Department of Civil and Environmental Engineering, University of Auckland, Auckland 1010, New Zealand (e-mail: gsen299@aucklanduni.ac.nz).

Ruggiero Lovreglio is with the School of Built Environment, Massey University, Auckland 0745, New Zealand.

Lukas Schubotz and Emile Chappin are with the Department of Technology, Policy and Management, Delft University of Technology, 2628 Delft, The Netherlands.

Digital Object Identifier 10.1109/TCSS.2025.3628208

GP as an explainable ABM tool that yields readable rules consistent with plausible individual behaviors. Historically, joint structure–parameter discovery followed sequential workflows or evolve structures under fixed parameters, then apply post-hoc calibration to a small set of finalists [5], or restrict models to a limited constant set embedded in expressions [6]. Although bilevel optimization (every candidate solution undergoes parameter optimization sequentially) would be ideal, it is computationally prohibitive [2]. As a middle ground, Vu et al. proposed a grammar-based "modest advance" that coevolves structures with embedded constants (bounded within $[0, 1]$) while allowing selection from promising precalibrated parameter sets, improving feasibility without full per-candidate calibration [6]. They also highlight that achieving optimal joint calibration will likely require efficient bilevel methods and surrogate models capable of estimating the performance of parameter-calibrated structures without exhaustive optimization [6].

B. Hybrid Approaches for Joint Optimization of Model Structure and Parameters

Hybrid evolutionary reinforcement learning (ERL) methods jointly optimize structures and parameters by combining evolutionary global search with gradient-based updates. For example, NEAT [9] and HyperNEAT [10] evolve both neural network topologies and weights, while more recent methods leverage Cartesian genetic programming (CGP) or grammatical evolution (GE) to discover symbolic policies with interpretable structures [11], [12]. Other studies have applied evolutionary feature synthesis [13] or human-in-the-loop symbolic regression [11] to derive compact control laws directly from neural network agents. While sharing the goal of joint optimization, these methods target forward control problems, learning policies that maximize cumulative reward, whereas ADOPT-GP addresses the inverse problem of discovering behavioral rules that reproduce observed macropatterns in multiagent systems. This implies different evaluation criteria (statistical fit versus reward maximization), resource management (dynamic allocation for expensive simulations versus fixed budgets), and integration strategies (simultaneous rule–parameter evolution versus staged optimization). Looking ahead, IGSS could extend beyond rule discovery to evolve whole ABM architectures (interaction topologies, environment dynamics, multilevel rules), a scope that exceeds typical ERL formulations.

C. Gaps in Existing Research

Existing ABM discovery methods face three issues: 1) decoupled structure–parameter optimization yielding suboptimal models; 2) unguided GP prone to bloating and noise; and 3) inefficient parallel evaluation. Bloating inflates complexity and cost [5], while stochasticity in GP/ABMs risks overfitting [14]. ADOPT-GP addresses these issues by coevolving structures and parameters in a unified dual loop, incorporating adaptive operator selection and symbolic regression for enhanced interpretability. A logistic-regression emulator provides fast parameter estimates during structural optimization, with promising candidates then fully calibrated via evolutionary algorithms, where dynamic CPU allocation improves smooth parallel processing. The result is interpretable, well-tuned rules outperforming fixed or sequential tuning workflows.

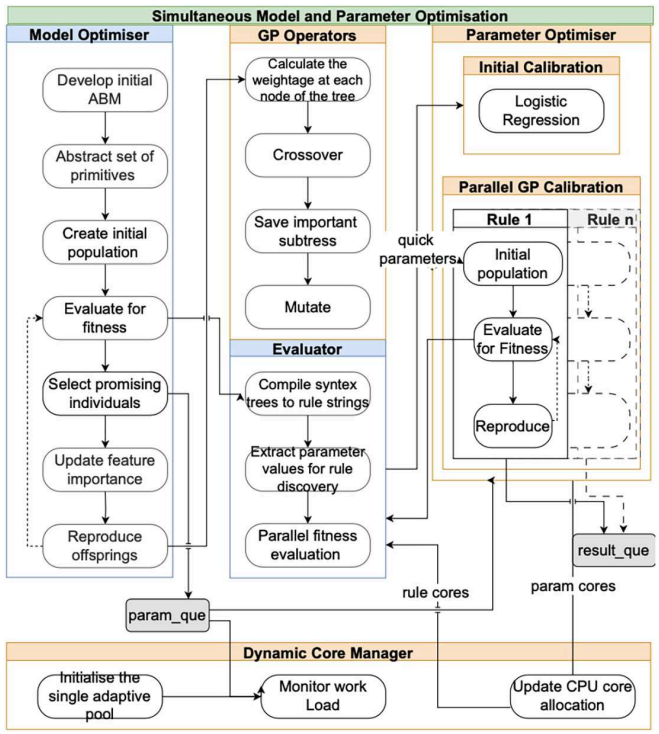


Fig. 1. ADOPT-GP framework overview. Adaptive GP evolves rule structures, with promising candidates undergoing fast logistic regression followed by detailed parameter calibration. Novel contributions (in yellow) include importance-based subtree selection, parallel parameter tuning, and adaptive thread allocation.

III. Methodology

A. Overview of the Proposed Framework

The ADOPT-GP framework integrates adaptive GP to simultaneously evolve ABM rule structures and calibrate parameters. As shown in Fig. 1, it comprises three components: 1) a structural discovery engine using adaptive operators; 2) a two-stage parameter optimizer combining quick logistic regression with full evolutionary calibration; and 3) a dynamic resource manager that allocates computational resources between structure and parameter search.

B. ABM Initialization and Data

1) Data: This study uses preevacuation data from two unplanned evacuations (false alarms) involving 253 students [15]. Prior analysis highlighted spatial location, social grouping, and item collection as key drivers. We transformed the dataset into a panel format to track individual decisions over time and expanded the feature set to include urgency and social influence. Microlevel data support initial parameter calibration, while macrolevel evacuation-time distributions serve as fitness targets for both rule structures and GP-based parameter calibration.

2) ABM Initialization: We implement a Python ABM to simulate preevacuation decision timing in a five-storey library. Agents (students) are heterogeneous, with attributes derived from data (e.g., gender, floor, group membership, and item collection propensity), reflecting the observed sample (55% female; 70% collectors). Each agent's evacuation decision is probabilistic, based on a utility function that combines social and psychological factors (Section III-C). Transition from the normal state (NS) to evacuation state (ES) follows a logistic function

$$p_{\text{evac}} = \frac{1}{1 + \exp(-U_{a,i})}. \tag{1}$$

Higher utility yields a higher likelihood of evacuation, with stochasticity preserving behavioral variability. The model outputs individual evacuation start times, which are compared to empirical distributions during GP-based optimization.

C. Hypothesized Causal Factors

IGSS benefits from theory-guided primitives and operators. While the goal is to "grow" explanations [2], model elements can be posited when grounded in theory. Following *Agent_Zero* [2], our agents combine affective (urgency), deliberative (rational cues), and social (peer influence) components. Table I lists the primitive factors and GP operators used to evolve evacuation rules. Factors map to core constructs in the theory of planned behavior (TPB), including attitudes (urgency), subjective norms (social influence), and perceived control (distance, item collection), which are operationalized as measurable variables [16]. PADM's cue interpretation and perceived threat are captured via a sigmoid transform of alarm urgency, reflecting bounded rationality and diminishing sensitivity (Weber–Fechner) [17]. Social influence and group effects are computed from nearby peers and in-group members, with parameters differing by normal versus evacuating states, drawing on social impact, social identity, and social comparison theories [18], [19]. Operators (addition, multiplication, division, negation, and exponentiation) compose these primitives into symbolic utility functions, which are converted to evacuation probabilities via a logistic link.

D. Adaptive Structural Optimization

ADOPT-GP implements an adaptive GP approach that dynamically adjusts its search strategy based on feature importance analysis and structural complexity, addressing common GP challenges such as bloating and noise learning. Complexity in this study is measured in two complementary ways (refer to Fig. 2), as follows.

1) *Tree Size*: The total number of nodes in the symbolic expression tree (including all functions and terminals). This metric is plotted in Fig. 4, where higher values indicate more complex, nested rules.
2) *Tree Depth*: The maximum path length from the root node to any leaf node. Tree depth serves as a secondary complexity constraint in the adaptive crossover and mutation operators.

TABLE I
AGENT_ZERO FRAMEWORK FACTORS

Primitives	Equation	Description
Affective Module		
Urgency (F_u)	$\frac{1}{1+e^{-(e^{\beta_1 F_t})}}$	Alarm urgency increasing over time
Deliberative Module		
Collecting (F_c)	Binary (0 or 1)	Decision to collect items
Time (F_t)	$time/max\_steps$	Time progression ratio
Ground Floor (F_{gf})	Binary (0 or 1)	Agent on ground floor
Distance (F_d)	$\lvert floor\_number \rvert/3$	Distance to exit (normalized)
Social Module		
Social Influence (F_{si})	$\left(\frac{nearby\_es}{total\_nearby}\right)^{\beta_2}$	Influence from nearby evacuations
Group Influence (F_{gi})	$\left(\frac{group\_es}{group\_size}\right)^{\beta_2}$	Influence from group evacuations
Group Size (F_{gm})	$group\_size$	Number of group members
In Group (F_g)	Binary (0 or 1)	Agent in a group
Operators		
Combine	$a + b$	Addition
Multiply	$a \times b$	Multiplication
Divide	$a/(b + \epsilon)$	Division (avoid zero)
Negate	$-a$	Negation
Exponential	e^a	Exponential of factor a

Note: β_1, β_2 calibrated during evolution.

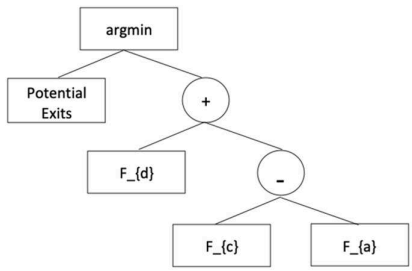


Fig. 2. Illustration of a symbolic expression tree used to define decision rules. The node count (expression size) is 6, representing all operators and terminals. The tree depth is 4, corresponding to the longest path from the root to any leaf.

1) Fitness Function: The performance of each model generated is evaluated by comparing the simulated preevacuation time distribution against empirical preevacuation data from the university library case study. Model fitness is assessed using two complementary metrics, as follows.

a) Kolmogorov–Smirnov (KS) statistic, measuring the maximum difference between simulated and empirical cumulative distributions. This metric captures differences in overall distribution shape without assuming a particular distribution type [20].
b) Kullback–Leibler (KL) divergence, quantifying the information loss when approximating the empirical

distribution with the simulated one, thereby identifying finer differences between distributions [21].

These metrics are combined into a single fitness score

$$f = D_{\mathrm{KS}}(S, E) + D_{\mathrm{KL}}(S||E) \tag{2}$$

where $D_{\mathrm{KS}}(S, E)$ is the KS statistic and $D_{\mathrm{KL}}(S||E)$ is the KL divergence. Lower fitness scores indicate a closer alignment between simulated and observed evacuation behaviors. The final fitness value for each model is computed by averaging across all ten simulation runs to ensure robustness against random variations.

2) Adaptive Crossover and Mutation Operator Modifications: ADOPT-GP steers evolution toward interpretable structures using adaptive crossover and mutation guided by feature importance and subtree depth. Feature importance is periodically updated using permutation importance from a random forest, and a compound score for each subtree is calculated as

$$S(t) = I(t) \cdot \left(1 - \frac{\text{depth}(t)}{\text{parent\_depth}}\right) \tag{3}$$

where $I(t)$ is the subtree's importance based on behavioral features. This scoring penalizes deep subtrees and highlights structurally compact, informative components.

During reproduction, crossover prioritizes exchanging subtrees with high $S(t)$ scores, drawn from a bounded repository (max size 30) of previously effective subtrees (Algorithm 1). Mutation targets low-importance subtrees and replaces them with stored or newly generated high-scoring alternatives (Algorithm 2). Only subtrees of moderate depth (2–3) are stored to balance expressiveness and simplicity.

The adaptive crossover and mutation strategies in Algorithms 1 and 2 address three major GP challenges in ABM discovery: 1) model bloat—excessively complex trees with no fitness gain; and 2) loss of critical behavioral components from random subtree exchanges. To mitigate these, each potential crossover or mutation point is scored using a composite metric that combines feature importance and a depth-based penalty. Only moderately sized subtrees (depth 2–3) are stored in a bounded repository, favoring interpretable yet expressive components. During reproduction, selection is biased toward compact, high-value subtrees, enabling targeted reuse and reducing disruptive recombination. This approach steers the evolutionary process toward simpler, more interpretable, and behaviorally meaningful rules while managing complexity.

Algorithm 1: Adaptive Crossover With Subtree Repository.

Require: Parent trees T_1, T_2, Feature importance scores I
Ensure: Modified trees after crossover
Constants: MAX_REPOSITORY_SIZE = 30 per return type
for each tree $T \in \{T_1, T_2\}$ **do**
  Identify valid crossover points P_T {All nodes except the root (to maintain tree integrity)}
  Calculate parent tree depth D_T
  for each point $p \in P_T$ **do**
    Extract subtree S_p at point p
    Calculate depth d_p of S_p
    Compute feature importance sum i_p of all features in S_p {Importance measures how critical the subtree's variables are}
    Compute depth penalty $penalty \leftarrow d_p/D_T$ {Penalise deeper subtrees to control bloat}
    Compute combined score $score_p \leftarrow i_p \cdot (1 - penalty)$ {Higher importance and lower depth yield higher scores}
    if $2 \leq d_p \leq 3$ **then**
      {Only store subtrees of moderate depth for reuse}
      if repository size ¡ $MAX\_REPOSITORY\_SIZE$ **then**
        Store S_p in repository
        Store $score_p$ to scores
      else
        Find the lowest-scoring subtree in the repository
        if $score_p > scores[lowest - scoringsubtree]$ **then**
          Replace lowest-scoring subtree with S_p
          Update score to $score_p$
        end if
      end if
    end if
  end for
  Normalise all $score_p$ into probabilities w_T {Higher-scoring subtrees are more likely to be selected}
end for
Randomly select crossover points p_1, p_2 using weights w_{T_1}, w_{T_2}
Extract subtrees S_{p_1} and S_{p_2}
Swap subtrees S_{p_1} and S_{p_2} in the parent trees
return T_1, T_2

E. Parameter Optimization

1) Quick Parameter Estimation Using Logistic Regression: During structural evolution, each candidate rule undergoes logistic regression using microlevel evacuation data [22], [23] to estimate coefficients for behavioral features. The intercept is embedded in the rule structure. This fast approximation provides informed initial parameters, accelerating subsequent optimization by seeding the population with plausible values.

2) Continuous Parameter Optimization Process: Promising candidates (those with above-average fitness and unique structures) are selected for detailed optimization (Algorithm 3). A historical archive avoids redundant optimization of structurally similar but inferior rules. Optimization is performed in parallel using dynamic thread allocation (Algorithm 4), covering both model and rule parameters. Latin hypercube sampling efficiently explores parameter spaces, guided by initial estimates from the quick stage.

A key challenge in ADOPT-GP is managing the computational load from parallel structure and parameter optimization. Algorithm 3 addresses this via a nonblocking, asynchronous workflow: parameter tasks are submitted to a parallel optimizer, returning future objects for independent execution and result tracking. Upon completion, calibrated rules are automatically processed and reintegrated into the structural evolution loop. Decoupled queues (Q_{param}, Q_{results}, Q_{feedback}) and controlled polling intervals ensure efficient CPU use without bottlenecks. This design strengthens joint optimization by continuously feeding calibrated parameter insights back into structural evolution.

F. Adaptive Evolutionary Process and Selection Strategy

Each ADOPT-GP run performs simultaneous evolution of rule structures and parameters. We report the final outputs (Table II) from three linked runs, where each is initialized with

Algorithm 2: Adaptive Mutation With Subtrees From Repository.

Require: Tree T, Feature importance scores I, Primitive set P, Repository R
Ensure: Modified tree after mutation
Identify valid mutation points P_T
for each point $p \in P_T$ **do**
  Calculate feature importance sum i_p in subtree at p
  $score_p \leftarrow 1 - i_p$ {Inverse importance}
end for
Normalise scores to probabilities w
Select mutation point p using weights w
$type \leftarrow$ return type at point p
if R contains subtrees of type $type$ **then**
  $S \leftarrow$ subtrees of type $type$ in R
  $H \leftarrow$ Scores of S
  $total \leftarrow \sum H$
  if $total > 0$ **then**
    $probs \leftarrow H_i/total$ for each score i
    $S_{new} \leftarrow$ select subtree from S using $probs$
  end if
end if
Replace subtree at p with S_{new}
return T

Algorithm 3: Parallel Rule Processing and Result Collection.

Require: Parameter queue Q_{param}, Results queue $Q_{results}$, Feedback queue $Q_{feedback}$
Ensure: Optimised rules with calibrated parameters
while optimisation is running **do**
  if Q_{param} not empty **then**
    $rule\_data \leftarrow Q_{param}.get()$ {Retrieve the next candidate rule needing parameter optimisation}
    $future \leftarrow$ SubmitOptimisationTask($rule\_data$) {Submit the rule to the parallel optimiser, returns a future object}
    AddCallback($future$, HandleResult) {Attach a callback so that when optimisation completes, results are handled automatically}
  end if
  {Process any completed optimisation results waiting in the results queue}
  while $Q_{results}$ not empty **do**
    $result \leftarrow Q_{results}.get\_nowait()$ {Retrieve the next completed optimisation result without blocking}
    $calibrated\_rule \leftarrow$ CreateCalibratedRule($result$) {Transform the raw optimisation output into a calibrated rule object}
    $Q_{feedback}.put(calibrated\_rule)$ {Place the calibrated rule into the feedback queue for use in the main evolutionary process}
  end while
  Sleep(monitoring_interval) {Pause briefly before the next polling cycle to avoid busy waiting}
end while

Algorithm 4: Dynamic Core Allocation With Hysteresis Control.

Require: Total available CPU cores N, Monitoring interval $t_{monitor}$, Hysteresis threshold θ
Ensure: Adaptive allocation of CPU threads between rule and parameter optimisation
Initialise core allocation: rule cores $c_r \leftarrow N-2$, parameter cores $c_p \leftarrow 2$
Create independent thread pools P_r and P_p with initial sizes c_r and c_p
Store initial allocation as $A_{\text{prev}} \leftarrow (c_r, c_p)$
Start monitoring thread
while optimisation is running **do**
  $q_{\text{param}} \leftarrow$ Number of active parameter tasks
  if $q_{\text{param}} \leq 2$ **then**
    $c_p \leftarrow 2$
  else if $q_{\text{param}} \leq 8$ **then**
    $c_p \leftarrow \max(2, \lfloor 0.25N \rfloor)$
  else if $q_{\text{param}} \leq 16$ **then**
    $c_p \leftarrow \max(2, \lfloor 0.5N \rfloor)$
  else
    $c_p \leftarrow \max(2, \lfloor 0.75N \rfloor)$
  end if
  $c_r \leftarrow N - c_p$
  $A_{\text{new}} \leftarrow (c_r, c_p)$
  $\delta \leftarrow \frac{|c_p - A_{\text{prev}}.param|}{N}$
  if $\delta > \theta$ **then**
    Update thread pools P_r, P_p with new sizes c_r, c_p
    $A_{\text{prev}} \leftarrow A_{\text{new}}$
  end if
  Wait $t_{monitor}$ seconds
end while

randomly generated individuals and five elite rules (those with the lowest fitness) retained from the previous run. This elite inheritance promotes inter-run refinement beyond stochastic variation.

Initial GP settings include a population of 30 individuals, a rule depth of 2–8, a mutation rate of 0.1, a crossover rate of 0.8, and a 50-generation limit. Each candidate rule is integrated into the ABM and evaluated using ten simulation replicates, with fitness scores averaged.

To improve efficiency, ADOPT-GP dynamically reduces population size by five whenever fitness improves over the global best, with a minimum size of 10. An early-stopping criterion halts evolution if no improvement is seen for ten generations [24].

After all runs, evolved candidates are pooled, and the five best-calibrated rules are selected for final analysis. This approach ensures robust convergence through both intra-run optimization and cross-run knowledge transfer.

G. Computational Efficiency Considerations

A major challenge in simultaneously optimizing agent rule structures and parameters lies in effectively managing computational resources across competing tasks. To address this, ADOPT-GP integrates a *dynamic core manager*, which reallocates CPU cores between rule evolution and parameter calibration based on the number of active optimization tasks, and a *dynamic optimization manager*, which periodically assesses workload and applies hysteresis control—only triggering reallocation when workload changes exceed a threshold (default $\theta = 0.1$ or 10%).

The system uses two dedicated thread pools for rule evolution and parameter optimization, dynamically adjusting their sizes based on the real-time number of running parameter tasks. Under light load (≤ 2 tasks), cores prioritize rule evolution; under balanced load ($\leq$ 8–16 tasks), resources are evenly distributed;

TABLE II
BEST CANDIDATE RULES WITH ORIGINAL AND CALIBRATED FITNESS VALUES

Run	Gen	Rule	Rule Detail	Original Fitness	Calibrated Fitness
2	1	Rule 1	$U = \frac{-0.69F_d}{e^{3.91F_t}} - 4.25$	0.434	0.176
1	6	Rule 2	$U = 3.27F_{gm}^2[0.05F_{si}(0.27F_u + 3.62F_t \cdot 0.27F_u \cdot e^{0.86F_{gf}} \cdot 0.22F_d) + 0.05F_{si} - 0.06F_{gm}] - 4.74$	0.269	0.184
1	20	Rule 3	$U = 0.08F_{gm}^2[0.82F_{si}(1.47F_u + 1.47F_u \cdot 0.55F_d + 0.82F_{si} \cdot e^{\frac{0.55F_d/0.82F_{si}}{0.55F_d+0.82F_{si}}} \cdot (-0.91F_{gm}))] - 4.22$	0.224	0.187
0	7	Rule 4	$U = \frac{(-0.55)^3 F_{gm}^3 (0.09F_u - 0.55F_{gm})}{0.09F_u e^{5.85F_t}} - 3.71$	0.265	0.188
1	26	Rule 5	$U = 0.09F_{gm}^2[0.44F_{si}(1.60F_u + 1.60F_u \cdot 0.62F_{gm} \cdot e^{0.44F_{si}} + 0.44F_{si} + 0.62F_{gm})] - 4.66$	0.222	0.191

and under heavy load (> 16 tasks), most cores are allocated to parameter tuning. Reallocation is triggered only when changes exceed the hysteresis threshold, minimizing overhead from frequent resizing. Rather than recreating thread pools, ADOPT-GP updates its sizes inline by modifying the `max_workers` attribute, enabling responsive and efficient load balancing that outperforms fixed-core strategies.

IV. RESULTS

The results discuss the performance improvements achieved by ADOPT-GP, discovered behavioral rules, and computational efficiency gains from dynamic resource allocation.

A. Evaluation of Model Discovery and Optimization Performance

1) Structural Optimization: Fig. 3 shows fitness convergence and complexity growth (tree size) across three evolutionary runs. All runs demonstrate rapid early-stage fitness improvements, reducing error from around 0.7 to below 0.25 within five generations. Run 2 achieves consistent low fitness (approximately 0.19) and stabilizes earlier, indicating effective optimization through elite population inheritance.

Tree size shown in Fig. 3(b) differs between runs. Run 0 gradually reaches stable complexity growth around 15. In contrast, runs 1 and 2 quickly stabilize (run 1 at 15 and run 2 at 25) due to elite population transfer, demonstrating the algorithm's ability to manage structural growth efficiently while preventing unnecessary complexity increases (bloating).

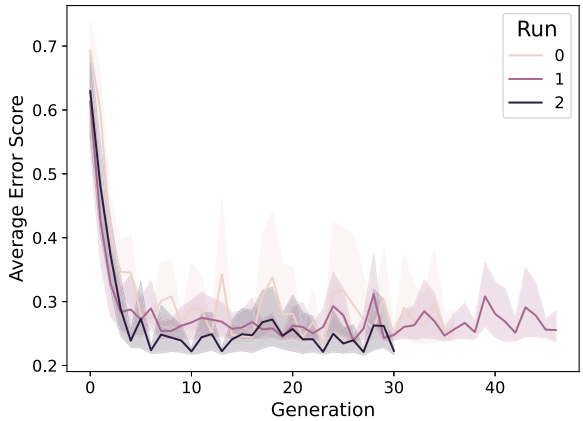
Fig. 4 further illustrates this relationship by plotting fitness and complexity growth of the best individuals in each generation. Key observations include the following.

1) 20% fitness improvement from run 0 (fitness = 0.22) to run 2 (0.19), indicating progressive optimization across runs.
2) Reduced fitness variability in later runs, highlighting algorithm stability.
3) Optimal tree size (25) in run 2, balancing sophisticated rule structures and improved predictive performance.

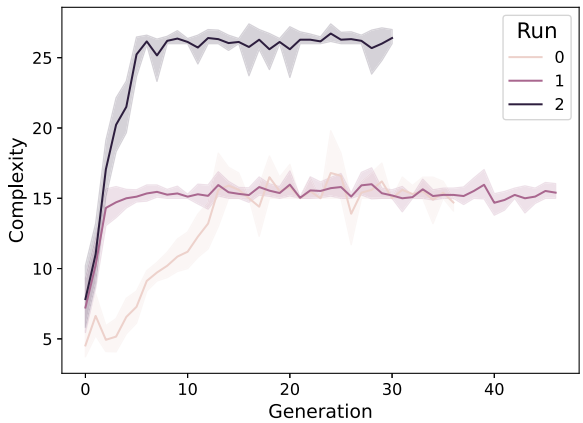
These results demonstrate that adaptive GP mechanisms to overcome unnecessary bloating and learning from noise, combined with elite population transfer, efficiently enhance model accuracy and complexity management across evolutionary runs.

Fig. 5 illustrates the consistency of convergence trends across adaptive GP runs. While the first run required additional generations to reduce fitness error, subsequent runs began with inherited elite solutions and stabilized quickly, demonstrating repeatability of the optimization strategy.

To further assess the robustness of the reported improvements, we conducted additional analyses across the three independent ADOPT-GP runs. Fig. 5 presents the 95% confidence intervals of the mean best fitness, illustrating consistent convergence across runs. While variance remained low in later



(a) Average error over generation



(b) Rule complexity (number of nodes) over Generation

Fig. 3. Evolution of model discovery metrics across generations. The shaded area represents the standard deviation across all individuals in the population at each generation. (a) Average fitness error over 50 generations for three evolutionary runs (run 0, run 1, and run 2). Lower fitness indicates better alignment with observed evacuation data. (b) Tree size over generations, showing how the algorithm controls bloating. Runs 1 and 2 use elite population transfer, resulting in faster stabilization.

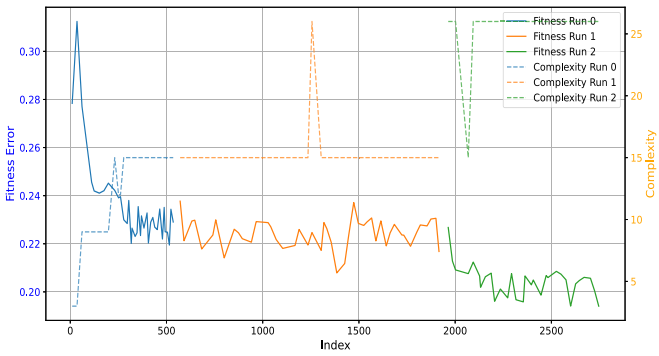


Fig. 4. Fitness error and tree size of the best-performing candidate rule across generations. The left y-axis shows fitness error (lower is better), while the right y-axis shows the number of nodes. Each line represents a different evolutionary run.

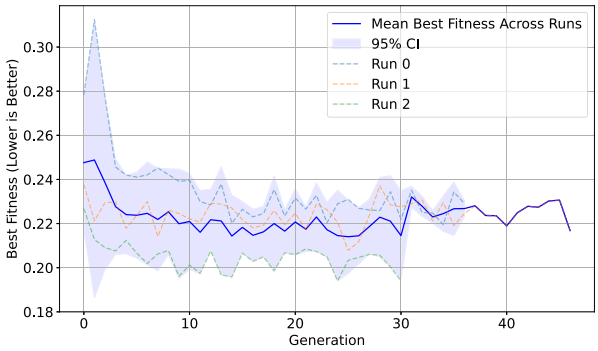


Fig. 5. Inter-run convergence consistency. The solid blue line shows the mean best fitness across three sequential adaptive GP runs, with the shaded area indicating the 95% confidence interval. Runs 1 and 2 started with elite individuals from the preceding run, resulting in lower initial fitness.

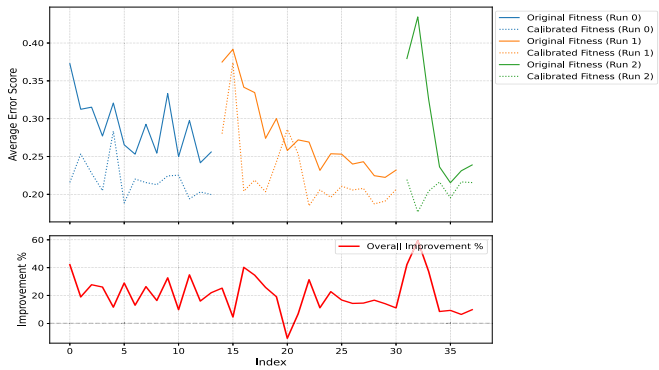


Fig. 6. Fitness error of promising candidates before and after parameter optimization. Continuous line shows original (preoptimization) fitness and dotted line shows calibrated (postoptimization) fitness for each run. Error reduction line in red demonstrates the benefit of dynamic parameter tuning.

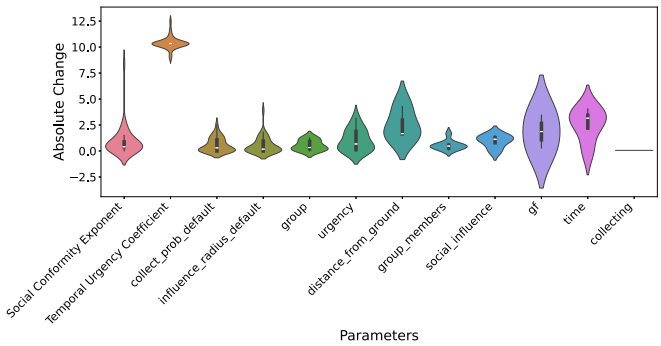


Fig. 7. Absolute change in parameter values after optimization. Each "violin" represents a specific parameter, illustrating the magnitude of adjustments during calibration. Behavioral parameters such as SCE and TUC show the highest variability.

generations, we acknowledge that stochastic effects can influence early-stage fitness trajectories. All runs converged to similar final fitness, demonstrating the robustness of the evolutionary process.

2) Parameter Optimization: Fig. 6 illustrates fitness improvements achieved through continuous parameter optimization of promising rules. Promising candidates were selected based on their above-average performance and structural uniqueness, preventing redundant optimizations.

Parameter optimization consistently improved fitness, with an average error reduction of 28.4%. Maximum improvements of up to 60% occurred early in generations, typically stabilizing between 20% and 40% during mid-stages, and reducing further (5%–15%) as rules approached optimality. Occasional cases of negative improvement reflect optimization challenges such as convergence to local minima or structural limitations within certain rules.

This finding challenges traditional IGSS practices that rely on fixed parameters, suggesting that valuable rule structures may be prematurely discarded without dynamic calibration.

Fig. 7 highlights significant variability in parameter adjustments. Parameters influencing behavioral dynamics, such as the social conformity exponent (SCE) and temporal urgency coefficient (TUC), exhibited wide variations, reflecting their sensitivity and importance in evacuation modeling. Conversely, structural parameters (collection probability, influence radius) showed limited adjustments due to strong empirical grounding. Coefficients for behavioral features (distance from ground, social influence, and time) often required nonlinear adjustments, emphasizing the complexity of interactions within evacuation decisions.

Overall, these patterns indicate the benefit of dynamic, data-driven parameter calibration over fixed initial parameterisations, especially during early and mid-stages of structural discovery.

3) Sensitivity of Fitness Aggregation Weighting: To assess whether the observed performance improvements depended on the specific weighting applied to the KS and KL metrics, we performed a sensitivity analysis by reevaluating the best discovered rule under alternative weighting schemes: 1:1 (equal), 1:3 (increased KL emphasis), and 3:1 (increased KS emphasis). As illustrated in Fig. 8, the distribution of fitness scores remained consistent across all configurations, with mean fitness values ranging narrowly from 0.178 to 0.206. The results demonstrate consistent performance and low variability across weighting

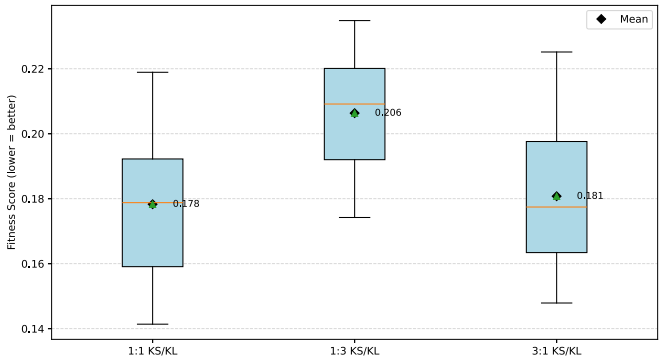


Fig. 8. Fitness sensitivity analysis across alternative KS/KL weighting schemes. Boxplots illustrate the distribution of fitness scores for the best discovered rule, evaluated across 20 independent simulation replicates under three weighting configurations: equal weighting (1:1), increased KL emphasis (1:3), and increased KS emphasis (3:1). Black diamonds denote the mean fitness values for each configuration.

schemes, supporting the robustness of the method's fitness evaluation.

B. Comparison Between Best Candidate Solutions

Table II summarizes the five best decision rules discovered across all three runs, comparing original and optimized fitness values. Rule 1 showed the greatest fitness improvement after calibration (from 0.434 to 0.176), which is also the simplest rule structure.

Each rule offers unique insights into preevacuation behavior mechanisms, which we analyze individually below, focusing on their theoretical implications and alignment with existing research on evacuation behavior.

1) Rule 1: This rule highlights the tension between physical proximity to safety (distance from the ground, F_d) and temporal decay ($e^{3.91F_t}$). The negative coefficient for F_d penalizes agents on higher floors, which aligns with empirical findings that vertical evacuation delays increase with floor height due to perceived risk and stairwell congestion [25]. The exponential decay term $e^{3.91F_t}$ reflects urgency dilution over time, consistent with protection motivation theory's "response cost" construct, where delayed action reduces perceived efficacy [26]. The intercept (-4.25) acts as a strong baseline hesitation, suggesting that agents require significant risk escalation to overcome inertia. This aligns with the "milling" phase in preevacuation behavior, where individuals seek confirmation before acting [27].

The simplicity of the rule, omitting social or group terms, implies evolutionary selection for individual risk-geographic prioritization over social dynamics, resonating with studies where isolated individuals prioritize vertical escape routes [28]. However, the lack of social influence contradicts herd behavior models, suggesting that this rule may dominate in low-crowd scenarios.

2) Rule 2: This rule emphasizes group size amplification (F_{gm}^2) and ground-floor proximity (F_{gf}), reflecting the emphasis of Social Identity Theory on in-group cohesion during crises [29]. The quadratic group term suggests that larger groups amplify collective risk assessment, consistent with the "fate sharing" behaviors observed in family evacuations [30]. The $e^{0.86F_{gf}}$ term prioritizes ground-floor agents, mirroring "familiarity bias" where individuals favor known exits [15]. The weak social influence coefficient (F_{si}) indicates evolutionary pressure to reduce social cues in favor of urgency-time interactions ($3.62F_t \cdot 0.27F_u$), aligned with adaptive decision-making under time constraints [31]. The negative group term ($-0.06F_{gm}$) introduces a paradoxical penalty for large groups, possibly reflecting evolutionary discovery of "coordination costs" in crowds [32]. The intercept (-4.74) indicates a strong baseline reluctance, requiring urgent, ground-floor-specific threats to trigger action. This rule encapsulates the tension between collective mobilization and logistical delays, echoing findings that group cohesion both aids and hinders evacuation efficiency [33].

3) Rule 3: This rule introduces nonlinear social influence via the term $e^{\frac{0.55F_d/0.82F_{si}}{0.55F_d+0.82F_{si}}}$, modeling diminishing returns of social cues as distance (F_d) increases. This aligns with networked threshold models [34], where social contagion weakens with physical separation. The urgency-distance interaction ($1.47F_u \cdot 0.55F_d$) reflects "risk compensation," where agents farther from exits heighten urgency to offset disadvantage, consistent with protection motivation theory's "threat appraisal" [26]. The negative group coefficient ($-0.91F_{gm}$) suggests evolutionary selection penalizes large groups in high-social-influence scenarios, possibly reflecting emergent "bottleneck" avoidance in dense crowds [35]. The intercept (-4.22) indicates moderate baseline hesitation, requiring urgent, socially validated threats to activate evacuation.

This rule mirrors dual-process theories, balancing rational risk-distance calculations with heuristic social mimicry, and aligns with empirical observations of "stampede" dynamics where social influence and spatial constraints interact nonlinearly [36].

4) Rule 4: This rule's cubic group penalty ($(-0.55)^3 F_{gm}^3$) and extreme time decay ($0.09F_u e^{5.85F_t}$) model "crowd stagnation," where large groups become paralyzed by indecision over time. The F_{gm}^3 term aligns with "social loafing" phenomena, where responsibility diffusion in groups delays action [37].

The numerator ($0.09F_u - 0.55F_{gm}$) suggests urgency only marginally offsets group inertia, consistent with studies showing preevacuation delays increase with group size [30]. The denominator $0.09F_u e^{5.85F_t}$ reflects rapid urgency decay, mirroring "temporal myopia" under stress, where agents discount long-term survival for short-term safety [38].

The intercept (-3.71) is the least negative, implying weaker baseline hesitation but extreme sensitivity to time-group interactions. This rule resonates with "frozen panic" scenarios, where crowded, time-pressured environments lead to collective inaction, and aligns with "bounded rationality" models where decision-making degrades under cognitive overload [39].

5) Rule 5: This rule emphasizes synergistic social-urgency interactions via $e^{0.44F_{si}}$, reflecting "critical mass" effects where social influence exponentially amplifies urgency. The $1.60F_u \cdot 0.62F_{gm}$ term models group-driven urgency escalation,

consistent with "emotional contagion" in cohesive groups [40]. The quadratic group term (F_{gm}^2) and linear social influence ($0.44F_{si}$) align with Social Impact Theory's strength-immediacy principles [18], where larger, closer-knit groups exert greater normative pressure. The intercept (-4.66) indicates strong baseline hesitation, requiring simultaneous social and urgent triggers to act. This rule mirrors "collective efficacy" frameworks [41], where group cohesion enhances perceived evacuation capability, and aligns with empirical findings that socially validated threats (for example, alarms + peer movement) accelerate response times [28]. The exponential term $e^{0.44F_{si}}$ suggests evolutionary discovery of nonlinear social tipping points, akin to "panic propagation" thresholds in crowd disasters [32].

6) Discovered Rules to Understand the Data and the Social System: The discovered rules collectively illustrate a complex interplay of spatial proximity, social dynamics, and urgency in preevacuation behavior, supporting existing evacuation theories and offering new quantitative insights.

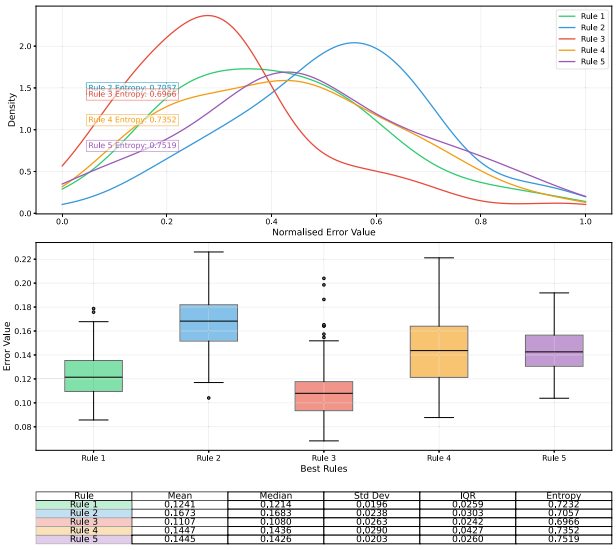
Group size significantly affects evacuation dynamics, presenting both advantages (enhanced collective urgency) and drawbacks (coordination delays). Negative coefficients and quadratic/cubic penalties formalize these conflicting effects. This duality aligns with Social Identity Theory's tension between in-group cohesion and logistical inefficiency [29], but extends it by formalizing group penalties as cubic terms (rule 4), a novel quantitative insight suggesting nonlinear deterioration of collective efficacy in dense crowds.

Spatial hierarchy is consistently prioritized over social factors; occupants strongly prefer known exits (on the ground floor) and penalize vertical distance, reinforcing known familiarity biases. Nonlinear effects frequently appear, capturing thresholds in social influence and perceptions of urgency. Terms like exponential social contagion (rule 5) and urgency decay (rule 4) indicate behavioral tipping points useful for evacuation planning. Consistent negative intercepts across rules confirm baseline hesitancy or "milling" behavior, requiring significant triggers (social or urgent cues) to initiate evacuation.

Robustness is evidenced by consistent patterns across rules. All intercepts are negative (-3.71 to -4.74), indicating evolutionary reinforcement of "baseline hesitation", consistent with real-world preevacuation delays [16]. Repeated parameters (e.g., F_{gm}^2 in rules 2/5 and e^3 in rules 2/5) suggest stable evolutionary selection for group-social interactions. The conflict between social influence (F_{si}) and spatial factors (F_d, F_{gf}) persists across rules, mirroring ABM studies where exit proximity mediates herd behavior [42].

While these evolved rules primarily formalize existing evacuation concepts, they reveal valuable hierarchies and interactions that can inform practical evacuation strategies. Further testing in different contexts (e.g., varied building types and crowd densities) will clarify the generalizability and limitations of these findings.

7) Robustness Analysis of the Best Candidates: The robustness of the best-performing rules was assessed using Gaussian Kernel entropy, selected for its flexibility in handling continuous error distributions [43]. Fig. 9 shows the performance consistency of the top five candidate rules across 100 simulations.



Rule	Mean	Median	Std Dev	IQR	Entropy
Rule 1	0.1241	0.1214	0.0196	0.0259	0.7232
Rule 2	0.1673	0.1683	0.0238	0.0303	0.7057
Rule 3	0.1107	0.1080	0.0263	0.0242	0.6966
Rule 4	0.1447	0.1436	0.0290	0.0427	0.7352
Rule 5	0.1445	0.1426	0.0203	0.0260	0.7519

Fig. 9. Error analysis of the five best candidate rules over 100 simulation runs. The density plot (top) shows the distribution of normalized error values. The table (bottom) summarizes the mean error, median error, standard deviation, interquartile range, and entropy, indicating the robustness and consistency of the predictions.

Rule 3 was the most robust, displaying the lowest entropy (0.6966), smallest mean error (0.1107), and moderate variability (IQR $= 0.0242$), indicating highly stable predictions.

Rule 2 had slightly lower robustness (entropy: 0.7057) and a higher average error (0.1673), although it was still consistent. Rule 1 performed competitively (entropy: 0.7232, mean error: 0.1241) with balanced variability. In contrast, rules 4 and 5 had higher entropy (0.7352 and 0.7519, respectively) and mean errors (0.145), suggesting less stable predictions.

Overall, rule 3 offers the optimal balance of accuracy and robustness. Rules 1–3 collectively represent a set of high-performing solutions, providing trade-offs between prediction accuracy and consistency, depending on modeling priorities.

C. Computational Performance

We assessed ADOPT-GP's dynamic core allocation using a 20-generation run with a population of 10. As shown in Fig. 10, the system exhibited three distinct phases: initially, few promising individuals led to low parameter demand and frequent core reallocations; mid-run, parameter task load increased, stabilizing allocation in favor of parameter tuning (24 parameter cores, 8 rule cores); and in the final phase, reduced demand shifted resources back to rule evolution. The core manager effectively tracked real-time workload based on the number of active parameter tasks and reallocated resources accordingly. Hysteresis control ensured stability by preventing frequent reallocation. While the current setup effectively maintained efficiency without contention, future improvements could include finer monitoring intervals or lower thresholds to better respond to short-lived queue spikes.

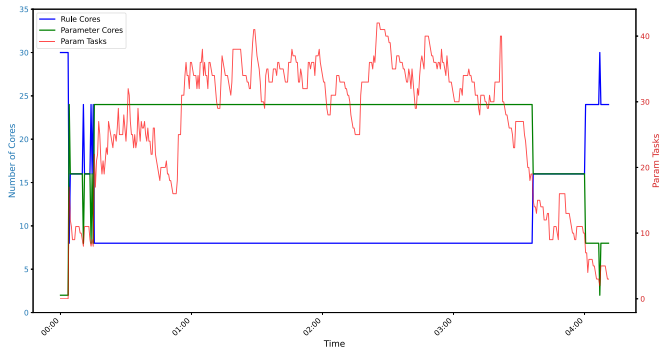


Fig. 10. Core allocation and parameter task queue dynamics over time during a single ADOPT-GP run. The system initially reallocates cores in response to increasing parameter load, then maintains a stable high-load allocation for most of the run, and finally shifts cores back to rule evaluation as parameter demand declines.

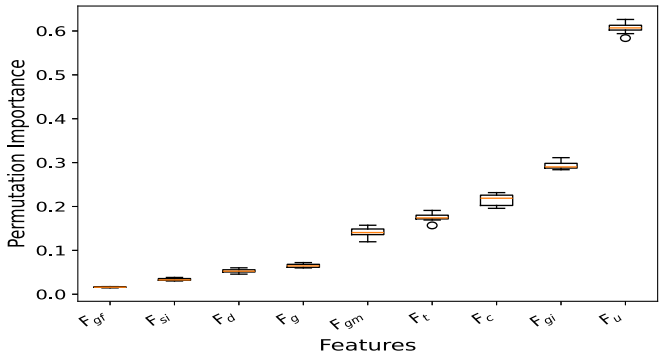


Fig. 11. Permutation importance of behavioral features in predicting evacuation timing. Higher values indicate greater influence on model fitness. Urgency (Fu) and group dynamics (Fgi, Fgm) are the most influential factors.

D. Feature Importance Analysis

The permutation importance analysis (Fig. 11) reveals a clear hierarchy of behavioral factors influencing preevacuation decisions. Urgency (F_u) emerges as the most influential factor (0.6), consistent with its central role in all evolved rules and real-world observations of increasing urgency during emergencies.

Group influence (F_{gi}) and collecting behavior (F_c) rank next (0.3 and 0.2), underscoring their significant contribution to evacuation decisions. Moderate importance (0.15) is observed for time (F_t) and group size (F_{gm}), indicating their secondary, supporting roles.

Lower ranked factors like social influence (F_{si}), distance from ground (F_d), and ground-floor status (F_{gf}) (below 0.1) initially appear contradictory due to their prominence in evolved rules. This reflects a known limitation of permutation importance, which is that it does not fully capture interactions between features. These factors primarily exert their influence through complex interactions—such as multiplicative terms with urgency or group factors—rather than in isolation.

Overall, this analysis highlights that preevacuation behavior is driven primarily by urgency and group dynamics, but accurate modeling requires considering both individual feature effects and their combined interactions.

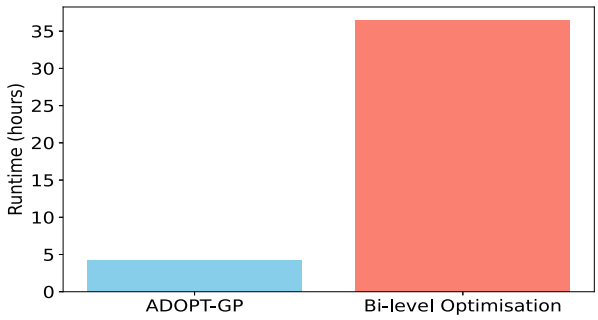


Fig. 12. Total runtime comparison between ADOPT-GP and bilevel sequential optimization. ADOPT-GP achieves an 85% reduction in runtime under identical experimental settings.

E. Comparison With Bilevel Sequential Optimization

To further address the source of performance gains and evaluate the architectural advantage of ADOPT-GP, we conducted a controlled comparison against a bilevel sequential optimization baseline. While the two approaches share a common objective of coevolving structure and parameters, they differ in their execution strategies.

In the bilevel setup, promising individuals (those with structurally unique rules and fitness below the generation mean) are sent to a nested parameter optimization loop after each generation completes. In contrast, ADOPT-GP evaluates promising individuals asynchronously in parallel, returning optimized versions to the main structural loop only if they show a fitness improvement of at least 20%. These optimized individuals are then retained in the main loop, guiding future evolution and enabling feedback-driven coadaptation.

Both methods were tested under identical experimental conditions, as follows.

1) 1 evolutionary run;
2) 20 structural generations;
3) 10 individuals per structural generation;
4) 5 individuals per parameter optimization generation;
5) 10 parameter generations;
6) Mutation rate $= 0.1$, Crossover rate $= 0.8$.

All hyperparameters, including the logistic regression initializer and simulation replicates, were kept constant across experiments to ensure fair comparison.

Fig. 12 shows the runtime comparison between ADOPT-GP and bilevel optimization. ADOPT-GP completes in approximately 5 h compared to over 36.5 h for the sequential method—nearly a 6.5× speedup and an 85% reduction in runtime.

In Fig. 13, each point represents a candidate rule generated during the run. Individuals with zero improvement are shown, as they were not sent for parameter optimization. In the case of ADOPT-GP, some zero improvement points may include candidates that did not achieve the 20% improvement threshold required for reintegration into the structural loop but were still fully optimized. Overall, ADOPT-GP achieves improvements in the range of 40%–80%, while bilevel optimization yields improvements primarily in the 30%–60% range.

These results clearly demonstrate that ADOPT-GP benefits not only from better parameter tuning but also from its

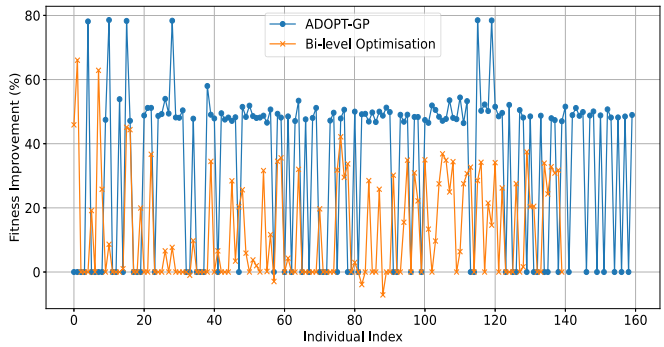


Fig. 13. Fitness improvement (%) of optimized individuals under ADOPT-GP and bilevel sequential optimization. ADOPT-GP achieves higher gains compared to bilevel optimization.

architectural advantages, including parallel integration, feedback-driven learning, and reduced redundancy and computational time.

V. Discussion

Compared to traditional IGSS workflows, which rely on fixed parameters or costly bilevel calibration, ADOPT-GP offers a practical middle ground—balancing fitness, interpretability, and efficiency. Unlike prior methods that either embed static constants [6] or the proposed bilevel optimization [2], [5], our approach enables candidate-specific parameter tuning during structural evolution. This results in up to 60% fitness improvement over fixed baselines, with an 80% reduction in runtime compared to bilevel alternatives.

ADOPT-GP's adaptive operator selection, guided by feature importance and rule complexity, mitigated bloating and maintained rule sizes between 15 and 25 nodes. This contrasts with traditional GP, where rules often grow excessively large. Future improvements could dynamically adjust subtree importance based on their impact on model fitness, similar to N-gram approaches in language processing.

The discovered rules revealed diverse behavioral strategies that trade off urgency, social influence, and spatial positioning. Some favor simple, predictable forms (e.g., urgency–distance), while others capture nonlinear social dynamics better suited to complex scenarios. This suggests context-dependent (rather than universal) decision logic and reinforces that behavior is situational and emergent. Still, these evolved rule expressions might be complex for nontechnical stakeholders to understand. The symbolic simplification or algebraic pruning can be used to reduce rule complexity and represent models as structured decision trees, thereby supporting effective communication.

Further, the ADOPT-GP algorithm is tested for a single evacuation scenario in a university library. Multidomain testing, such as high-density transit hubs, shopping centers, or culturally diverse populations, is a future area of work. Second, the evacuation psychology (panic, urgency under life threat, or uncertainty) remains challenging to represent in data-driven model discovery. Maintaining interpretability in ADOPT-GP is hard when discovery optimizes only for empirical fit; sense-making can erode even as fitness improves. A promising direction is to align candidate rules with established theories during the search. While expert review can provide this, it disrupts automation and is time-intensive; emerging LLM-based evaluators, on the other hand, offer a scalable way to inject theory checks and strengthen the grounding of evolved rules.

Other methodological advancements may include surrogate modeling techniques for parameter optimization and automatically generated building blocks/primitives rather than predefined ones [44]. Explore how ADOPT-GP can be adapted to domains with limited empirical data availability. Potential strategies may include transfer learning approaches and the use of hybrid datasets that combine observed data with synthetic or scenario-based data, incorporating variables that represent stress and urgency. While the algorithm demonstrates consistent convergence and low inter-run variance, some inherent variability remains in stochastic evolutionary processes. In future work, ensemble strategies combining predictions from multiple high-performing rules could further improve robustness and generalizability.

From a practical standpoint, ADOPT-GP lays the groundwork for decision support tools that integrate with building information systems. To translate computational insights into safety strategies, we recommend the following.

1) *Reducing Coordination Costs*: It designs wider stairways and decentralized exits for large-group evacuations.
2) *Utilizing Social Influence*: It combines alarms with directional visual cues to exploit social behavioral thresholds.
3) *Improving Exit Visibility*: It prioritizes clear signage on ground and upper floors to counteract distance-related hesitation.
4) *Counteracting Urgency Decay*: It implements escalating real-time alarms to sustain occupant urgency.

These strategies demonstrate how computational insights can effectively inform evacuation safety improvements, bridging theory, and real-world applications.

VI. Conclusion

This study presents ADOPT-GP, a novel framework for jointly optimizing ABM rule structures and parameters within the IGSS paradigm. Applied to a university library evacuation case, the method uncovered diverse, context-specific decision rules and achieved notable fitness gains with significantly reduced computational cost.

Key innovations include adaptive GP operators that promote interpretability, a dual-stage parameter calibration process, and dynamic parallel resource allocation. Together, these components enable efficient, feedback-driven model discovery.

Beyond methodological contributions, the discovered rules offer practical insights for emergency planning, such as designing interventions tailored to building layouts and occupant behaviors. Overall, ADOPT-GP advances inverse modeling by balancing behavioral realism, computational efficiency, and model plausibility—supporting more transparent, context-sensitive decision-making in complex social systems.

REFERENCES

[1] G. P. Senanayake, M. Kieu, Y. Zou, and K. Dirks, "Agent-based simulation for pedestrian evacuation: A systematic literature review," *Int. J. Disaster Risk Reduct.*, vol. 111, 2024, Art. no. 104705.

[2] J. M. Epstein, "Inverse generative social science: Backward to the future," *J. Artif. Soc. Social Simul.*, vol. 26, no. 2, 2023.

[3] V. Grimm and S. F. Railsback, "Pattern-oriented modelling: A 'multi-scope' for predictive systems ecology," *Philos. Trans. R. Soc. B*, vol. 367, no. 1586, pp. 298–310, 2012.

[4] R. Greig, C. Major, M. Pacholska, S. Bending, and J. Arranz, "Learning interpretable logic for agent-based models from domain independent primitives," *J. Artif. Soc. Social Simul.*, vol. 26, no. 2, 2023.

[5] C. Gunaratne, E. Hatna, J. M. Epstein, and I. Garibay, "Generating mixed patterns of residential segregation: An evolutionary approach," *J. Artif. Soc. Social Simul.*, vol. 26, no. 22023.

[6] T. M. Vu, C. Buckley, J. A. Duro, A. Brennan, J. M. Epstein, and R. C. Purshouse, "Can social norms explain long-term trends in alcohol use? Insights from inverse generative social science," *J. Artif. Soc. Social Simul.*, vol. 26, no. 2, 2023.

[7] J. Zhong, L. Luo, W. Cai, and M. Lees, "Automatic rule identification for agent-based crowd models through gene expression programming," in *Proc. Int. Conf. Auton. Agents Multi-Agent Syst.*, 2014, pp. 1125–1132.

[8] V. A. Smith, "Evolving an agent-based model to probe behavioral rules in flocks of cowbirds," in *Proc. ALIFE*, vol. 2008, 2008, pp. 561–568.

[9] K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evol. Comput.*, vol. 10, no. 2, pp. 99–127, 2002.

[10] K. O. Stanley, D. B. D'Ambrosio, and J. Gauci, "A hypercube-based encoding for evolving large-scale neural networks," *Artif. Life*, vol. 15, no. 2, pp. 185–212, 2009.

[11] G. Nadizar, L. Rovito, A. De Lorenzo, E. Medvet, and M. Virgolin, "An analysis of the ingredients for learning interpretable symbolic regression models with human-in-the-loop and genetic programming," *ACM Trans. Evol. Learn. Optim.*, vol. 4, no. 1, pp. 1–30, 2024.

[12] I. G. Tsoulos, A. Tzallas, and E. Karvounis, "Using optimization techniques in grammatical evolution," *Future Internet*, vol. 16, no. 5, 2024, Art. no. 172.

[13] H. Zhang, A. Zhou, and X. Lin, "Interpretable policy derivation for reinforcement learning based on evolutionary feature synthesis," *Complex Intell. Syst.*, vol. 6, no. 3, pp. 741–753, 2020.

[14] L. Miranda, O. O. Garibay, and J. Baggio, "Evolutionary model discovery of human behavioral factors driving decision-making in irrigation experiments," *J. Artif. Soc. Social Simul.*, vol. 26, no. 2, 2023.

[15] R. Lovreglio and E. Kuligowski, "A pre-evacuation study using data from evacuation drills and false alarm evacuations in a university library," *Fire Saf. J.*, vol. 131, 2022, Art. no. 103595.

[16] E. Kuligowski, "Predicting human behavior during fires," *Fire Technol.*, vol. 49, pp. 101–120, 2013.

[17] F. E. Cornes, G. A. Frank, and C. O. Dorso, "Fear propagation and the evacuation dynamics," *Simul. Modell. Pract. Theory*, vol. 95, pp. 112–133, 2019.

[18] B. Latané, "The psychology of social impact," *Amer. Psychol.*, vol. 36, no. 4, 1981, Art. no. 343.

[19] M. A. Hogg, *Social Identity Theory.* Springer, in *Understanding Peace and Conflict Through Social Identity Theory* (Peace Psychology Book Series), S. McKeown, R. Haji, and N. Ferguson, Eds., Cham: Springer, 2016. [Online]. Available: https://doi.org/10.1007/978-3-319-29869-6_1.

[20] F. J. Massey, Jr., "The Kolmogorov-Smirnov test for goodness of fit," *J. Am. Stat. Assoc.*, vol. 46, no. 253, pp. 68–78, 1951.

[21] R. Lovreglio, E. Ronchi, and D. Borri, "The validation of evacuation simulation models through the analysis of behavioural uncertainty," *Reliab. Eng. Syst. Saf.*, vol. 131, pp. 166–174, 2014.

[22] R. Lovreglio, E. Ronchi, and D. Nilsson, "A model of the decision-making process during pre-evacuation," *Fire Saf. J.*, vol. 78, pp. 168–179, 2015.

[23] R. Lovreglio, E. Ronchi, and D. Nilsson, "An evacuation decision model based on perceived risk, social influence and behavioural uncertainty," *Simul. Modell. Pract. Theory*, vol. 66, pp. 226–242, 2016.

[24] B. Zachariah, S. Misra, P. O. Odion, and S. R. Isah, "MRDPGA: A multiple restart dynamic population genetic algorithm for scheduling road traffic," *J. Electr. Syst. Inf. Technol.*, vol. 10, no. 1, 2023, Art. no. 35.

[25] J. L. Pauls, "Building evacuation: Research findings and recommendations," *Fires Hum. Behav.*, pp. 251–275, 1980.

[26] R. W. Rogers, "A protection motivation theory of fear appeals and attitude change1," *J. Psychol.*, vol. 91, no. 1, pp. 93–114, 1975.

[27] E. D. Kuligowski, "Human behavior in fire," in *SFPE Handbook of Fire Protection Engineering.* New York, NY, USA: Springer New York, 2016, pp. 2070–2114.

[28] G. Proulx and R. F. Fahy, "The time delay to start evacuation: Review of five case studies," *Fire Saf. Sci.*, vol. 5, pp. 783–794, 1997.

[29] J. Drury, C. Cocking, and S. Reicher, "The nature of collective resilience: Survivor reactions to the 2005 London bombings," *Int. J. Mass Emerg. Disasters*, vol. 27, no. 1, pp. 66–95, 2009.

[30] H. Mu, J. Wang, Z. Mao, J. Sun, S. Lo, and Q. Wang, "Pre-evacuation human reactions in fires: An attribution analysis considering psychological process," *Procedia Eng.*, vol. 52, pp. 290–296, 2013.

[31] R. Lovreglio, A. Fonzone, and L. Dell'Olio, "A mixed logit model for predicting exit choice during building evacuations," *Transp. Res. Part A: Policy Pract.*, vol. 92, pp. 59–75, 2016.

[32] D. Helbing, I. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic," *Nature*, vol. 407, no. 6803, pp. 487–490, 2000.

[33] N. Shiwakoti, M. Sarvi, G. Rose, and M. Burd, "Animal dynamics based approach for modeling pedestrian crowd egress under panic conditions," *Procedia-Soc. Behav. Sci.*, vol. 17, pp. 438–461, 2011.

[34] M. Wiedermann, E. K. Smith, J. Heitzig, and J. F. Donges, "A network-based microfoundation of Granovetter's threshold model for social tipping," *Sci. Rep.*, vol. 10, no. 1, 2020, Art. no. 11202.

[35] D. Helbing, L. Buzna, A. Johansson, and T. Werner, "Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions," *Transp. Sci.*, vol. 39, no. 1, pp. 1–24, 2005.

[36] X. Zheng, T. Zhong, and M. Liu, "Modeling crowd evacuation of a building based on seven methodological approaches," *Build. Environ.*, vol. 44, no. 3, pp. 437–445, 2009.

[37] B. Latané, K. Williams, and S. Harkins, "Many hands make light the work: The causes and consequences of social loafing," *J. Pers. Soc. Psychol.*, vol. 37, no. 6, 1979, Art. no. 822.

[38] G. Loewenstein and D. Prelec, "Anomalies in intertemporal choice: Evidence and an interpretation," *Quart. J. Econ.*, vol. 107, no. 2, pp. 573–597, 1992.

[39] H. A. Simon, "A behavioral model of rational choice," *Quart. J. Econ.*, vol. 69, pp. 99–118, 1955.

[40] S. G. Barsade, "The ripple effect: Emotional contagion and its influence on group behavior," *Administ. Sci. Quart.*, vol. 47, no. 4, pp. 644–675, 2002.

[41] A. Bandura, "Exercise of human agency through collective efficacy," *Curr. Dir. Psychol. Sci.*, vol. 9, no. 3, pp. 75–78, 2000.

[42] N. Shiwakoti and M. Sarvi, "Enhancing the panic escape of crowd through architectural design," *Transp. Res. C: Emerg. Technol.*, vol. 37, pp. 260–267, 2013.

[43] J. He, G. Wang, K. Cao, H. Diao, G. Wang, and B. Peng, "Generalized minimum error entropy for robust learning," *Pattern Recognit.*, vol. 135, 2023, Art. no. 109188.

[44] A. Collins, M. Koehler, and C. Lynch, "Methods that support the validation of agent-based models: An overview and discussion," *J. Artif. Soc. Social Simul.*, vol. 27, no. 1, 2024.

Gayani P. D. P. Senanayake is currently working toward the Ph.D. degree in transport resilience with the University of Auckland, Auckland, New Zealand, with a specific focus on developing a data-driven method to create agent-based models of evacuation systems, such as people exiting a train station.

Her research interests include agent-based modeling, inverse modeling, and evacuation behavior.

Minh Kieu received the Ph.D. degree in transport engineering from Queensland University of Technology, Brisbane, Australia, in 2019.

He is currently a Senior Lecturer with the Faculty of Engineering and Design, Department of Civil and Environmental Engineering, University of Auckland, Auckland, New Zealand. He is also the Future and Intelligent Transport Systems Lead with the Transport Research Centre, University of Auckland, Auckland, New Zealand.

His research interests include urban analytics, machine learning, artificial intelligence, agent-based modeling, computer simulation, data-driven systems, infrastructure systems, transport resilience, and traffic safety.

Ruggiero Lovreglio received the Ph.D. degree in civil engineering from the Inter-Polytechnic Doctoral School: Politecnico di Bari, Bari, Italy, Politecnico di Milano, Milan, Italy, and Politecnico di Torino, Turin, Italy, in 2016.

He is currently a Professor with Massey University, Auckland, New Zealand, and a Rutherford Discovery Fellow with the Royal Society Te Apārangi, Wellington, New Zealand. He is the Co-Chair of the International Association for Disaster Reduction and Emergency Management, founded by Tsinghua University, Beijing, China. His work focuses on human behavior in disasters, evacuation modeling, and safety training. My research makes use of new digital technologies such as artificial intelligence, virtual reality, augmented reality, and BIM.

Yang Zou is currently a Senior Lecturer in construction engineering and management with the Faculty of Engineering and Design, Department of Civil and Environmental Engineering, University of Auckland, Auckland, New Zealand. His research interests include the inter-disciplinary area of construction automation and robotics. Specifically, he has focused on applying digital twins (DTs), building information modeling (BIM), unmanned aerial systems (UASs), robotics, artificial intelligence (AI), extended reality (VR/AR/MR), smart sensing and visualization, throughout the lifecycle of complex buildings and large-scale infrastructure systems.

Kim Dirks is currently a Professor with the Faculty of Engineering and Design, Department of Civil and Environmental Engineering, University of Auckland, Auckland, New Zealand. She brings a multidisciplinary lens to both her teaching and research, having held academic appointments across science, medical and health sciences, and engineering. Her research focuses on the ways in which urban civil infrastructure impacts the health and well being of urban residents, in particular air pollution and noise from road transport. In air pollution, her work includes measuring human exposure using portable sensors and modeling air pollution levels in response to changes in local meteorology and transportation infrastructure. In the area of noise, her research focuses on the impact of road traffic and community annoyance. She also has interests in the role of urban greenspace in promoting community health and wellbeing.

Lukas Schubotz received the bachelor's degree in mathematics with a minor in psychology from Heidelberg University, Heidelberg, Germany, in 2019, and the master's degree in innovation science from Utrecht University, Utrecht, Netherlands, in 2023. He is currently working toward the Ph.D. degree in energy systems with the Energy Transition Lab, Delft, Netherlands, and the Energy and Industry Group, Faculty of the Department of Technology, Policy and Management, Delft University of Technology, Delft, The Netherlands.

His research focuses on developing inverse modeling techniques for agent-based models and methods for subsequent sense-making to study behavior in and related to the design of the energy transition. His work unravels and connects the computational challenges of inverse modeling and nested uncertainty and chaos within to the generative, interpretative domain of behavioral psychology, policy decision support, and scenario exploration.

Emile Chappin is currently an Associate Professor in energy systems and simulation and the Head of the Energy and Industry Group, Department of Engineering Systems and Services, Faculty of the Department of Technology, Policy and Management, Delft University of Technology, Delft, The Netherlands. He is also a Co-Director of the TPM Energy Transition Lab, Delft, Netherlands. He is known for his broad expertise in all energy systems, energy policy, and agent-based modeling. His research interests include the role of simulation modeling, in particular agent-based modeling, in the domain of energy systems. In his research, he connects developments in modeling methodology to an understanding of complex sociotechnical energy systems in order to better understand how to model, analyze, and design (parts) of the energy transition. His primary interests are to unravel the possible long-term dynamics of energy systems and how they are shaped by energy policies, and to contribute to the modeling methods that address important academic and societal questions in this area.