METROREHEARSAL: TOOL-GUIDED MULTI-AGENT DEBATE FOR METRO EMERGENCY PLANNING

Jinlin Li & Xiao Zhou *
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
{jinlin2021,xiaozhou}@ruc.edu.cn

Yingying Zhang & Xian Wu
Tencent Jarvis Lab
Beijing, China
{ninzhang,kevinxwu}@tencent.com

ABSTRACT

Major hubs in urban rail transit are prone to cascading failures when faced with temporary control measures, demand pulses induced by events or holidays, and operational interventions (e.g., increased headways and ad hoc timetable adjustments). Operating agencies therefore need actionable ex ante decision support to rapidly translate scenario specifications into executable contingency plans; however, existing methods either require substantial scenario modeling and engineering effort to capture fine-grained within-station processes, or still rely heavily on manual expertise, making it difficult to quickly distill reusable decision rules. To address this gap, we propose **MetroRehearsal**, a generative-agent framework for metro-hub emergency rehearsal that enables a rapid closed loop from scenario specification to plan output. Given an operational scenario and a disruption setting, MetroRehearsal generates structured candidate plans in parallel, queries external tools to verify feasibility and quantify costs for key diversion and alternative-station recommendations, and produces a final plan through evidence-grounded multi-agent debate with consensus convergence. We conduct experiments on real-world Shanghai metro data and network topology. Results show that, compared with single-shot generation or tool-only verification baselines, combining tool-augmented evidence with multi-agent debate jointly improves plan feasibility, cost efficiency, and decision stability.

1 INTRODUCTION

Major hubs in urban rail transit often face abrupt passenger-demand surges during holidays and large-scale events. When large volumes arrive within a short time window, concourse and platform densities can rise rapidly, triggering crowd spillover and cascading delays that may propagate across stations and lines Lei et al. (2025). Such peaks are frequently compounded by temporary control measures, including station, entrance/exit, or transfer-corridor closures, as well as operational interventions such as reduced service frequency and ad hoc timetable adjustments, which further amplify system instability. Therefore, operating agencies urgently need executable ex ante decision support tailored to such time-sensitive disruptions. Prior to onset, the system should take a specified control plan or event script as input, rehearse which high-risk hubs, transfer pairs, and time windows are most likely to trigger instability and spillover, and produce executable emergency response plans, including diversion guidance and alternative-station recommendations, to keep risks within a manageable range Wang et al. (2025).

However, existing methods still struggle to support this rapid closed loop from scripts to plans. On the one hand, large scale agent based transportation simulation frameworks, such as MATSim, can generate city scale flows and evaluate policy impacts, but they typically require additional modeling and scenario engineering to represent fine grained processes such as within station facility bottlenecks, transfer organization, and traveler responses W Axhausen et al. (2016); Yin et al. (2019). On the other hand, for typical disruptions such as station closures, prior studies have combined behavioral optimization with simulation to characterize passengers' substitution choices over routes and

*Corresponding author

stations under closure conditions and to assess the resulting demand impacts, indicating that closure rehearsal is methodologically well defined and computationally tractable Yin et al. (2016). Yet translating rehearsal outputs into reusable operational rules and frontline action guidance still relies heavily on manual expertise, and it is difficult to rapidly transfer across hubs with different demand structures and different combinations of disruptions.

Large language models (LLMs) and generative agents offer a new technical pathway for emergency rehearsal in metro systems Wu et al. (2023); Qian et al. (2024). By generating multiple candidate contingency plans in parallel and organizing structured debate, the system can improve decision robustness through "cross-examination and consensus." Du et al. (2023) Meanwhile, introducing external tools that provide verifiable evidence shifts plan comparison from subjective textual preferences to evidence-constrained selection, reducing speculative judgments about the feasibility and cost of diversion or alternative-station recommendations Yao et al. (2023). Motivated by these observations, we propose **MetroRehearsal**, a generative-agent framework for metro-hub emergency rehearsal. Given an operational scenario and a disruption specification, the framework generates structured candidate plans in parallel, queries a public-transit routing tool to verify feasibility and quantify costs for key diversion suggestions, and then performs evidence-grounded multi-agent debate with consensus aggregation. It outputs a finalized contingency plan together with a risk ranking and concise explanations.

Our key contributions include: (1) we formalize the metro-hub emergencyn rehearsal task and define an executable structured output specification; (2) we propose a plan generation framework that combines parallel candidate generation, tool-augmented evidence, and debate-and-consensus consolidation; and (3) we build rehearsal settings on real-world metro data and network topology and demonstrate the effectiveness of our approach.

2 Related Work

Multi-Agent Debate. In recent years, research on large language models (LLMs) has gradually expanded from single-agent generation paradigms to multi-agent systems. The central motivation of this line of work is to enhance the reliability and accuracy of the reasoning process through cooperative or adversarial interactions among multiple agents. Prior studies have shown that introducing multi-round interaction mechanisms allows agents to cross-check and correct errors during reasoning, thereby improving the consistency and robustness of logical inference Du et al. (2023); Chen et al. (2024). Beyond general reasoning tasks, multi-agent paradigms have also been applied to more structured task settings, such as mathematical reasoning and safety assessment. For example, in mathematical reasoning, debate-based self-correction strategies improve answer correctness by encouraging agents to scrutinize intermediate derivation steps Zhang & Xiong (2025). In safety alignment research, divergences among agents' perspectives are leveraged to expose potentially unsafe behaviors Asad et al. (2025). Overall, these studies demonstrate that multi-agent debate can enrich the reasoning process and produce more diverse reasoning trajectories, but its effectiveness still largely depends on the internal knowledge and capabilities of the underlying models.

Tool-Augmented Agents. In recent years, researchers have increasingly explored extending the capability boundaries of LLMs by incorporating external tools, enabling them to handle tasks that require specialized knowledge or complex computations. Early work such as Toolformer equips models with the ability to learn when and how to invoke APIs (Schick et al., 2023). As this line of research has progressed, tool usage has gradually evolved from simple interface calls toward more systematic agentic pipelines, which are typically organized in a modular fashion and integrate components such as information retrieval, symbolic reasoning, multi-step planning, and domain-specific simulators Bran et al. (2023); Wu et al. (2023); Webb et al. (2025); Yao et al. (2023).

Despite these advances, most existing tool-augmented frameworks remain methodologically centered on a single-agent perspective, where tools are typically invoked in a one-shot or sequential manner. Such designs have yet to fully exploit interactions among multiple agents for coordinating tool selection, result verification, and decision revision, which in turn limits their stability and robustness in complex decision-making scenarios.

Furthermore, in complex operational systems such as urban and transportation domains, research that tightly couples multi-agent LLMs with external planners or evaluators to form a verifiable de-

cision loop remains limited. Existing studies often treat language models primarily as analytical or control interfaces, without systematically embedding tool invocation into the agent reasoning process to support ex ante decision evaluation and scenario rehearsal.

3 PRELIMINARY

This work study a *metro-hub emergency rehearsal* task in which, given an operational scenario, the system generates an executable contingency plan and verifies it with an external evaluator. Formally, a scenario is denoted as

$$x = (\mathcal{N}, \mathcal{D}, \mathcal{S}), \tag{1}$$

where $\mathcal{N}$ is the metro network with station/transfer relations,

$\mathcal{D}$ denotes the baseline travel demand statistics aggregated from AFC (Automatic Fare Collection) records, which are used to characterize the passenger-flow distribution and spatiotemporal travel patterns of the metro network under undisturbed conditions, thereby providing a statistical description of passenger-flow structures across the entire network. (e.g., time-sliced station inflow/outflow volumes and OD (origin–destination) demand). $\mathcal{S}$ denotes the operational setting, which consists of a set of events, each parameterized in a unified *entity–time window–intensity* form:

$$\mathcal{S} = \{(\ell_j,\ [\tau_j^s, \tau_j^e],\ \xi_j)\}_{j=1}^{m}, \tag{2}$$

where ℓ_j denotes the affected entity (e.g., a station, an entrance or exit), $[\tau_j^s, \tau_j^e]$ is the effective time window, and ξ_j is the affected intensity.

We introduce two guidance actions, namely **diversion guidance** and **alternative-station recommendation**, denoted by $a \in \{a_{\text{div}}, a_{\text{alt}}\}$. These actions are designed to generate operationally feasible passenger-guidance plans under three representative types of emergency scenarios: (1) **Temporary closures** (closure or restricted access of stations, entrances/exits, or transfer corridors); (2) **Event- or holiday-induced demand surges** (specified stations experience a sharp demand increase during certain time windows); (3) **Service changes** (increased headways, reduced service frequency, temporary skip-stop operations, or one-directional transfer arrangements).

For any given scenario x, we assume that at least one action type in $\{a_{\text{div}}, a_{\text{alt}}\}$ satisfies the executability constraints, and each candidate plan instantiates only one action type. Specifically, a diversion action is defined as $a_{\text{div}} = (o, z, \tau)$, where o denotes the affected passenger-flow unit induced jointly by the disturbed entity ℓ_j and the baseline demand distribution $\mathcal{D}$, typically corresponding to OD demand or station-level passenger flows that would otherwise pass through ℓ_j. z denotes the diversion target, such as an alternative station or route, and τ denotes the operational time window $[\tau_j^s, \tau_j^e]$, within which the action is executed. Similarly, an alternative-station recommendation action is defined as $a_{\text{alt}} = (o, z', \tau)$, where o again denotes the affected passenger-flow unit, z' denotes the recommended alternative station, and τ denotes the effective time window.

Problem Setup. Given a scenario $x = (\mathcal{N}, \mathcal{D}, \mathcal{S})$, our goal is to generate a structured contingency plan y^* within the feasible action space $\mathcal{A}$. Let y denote a structured contingency plan, and let $\mathcal{Y}$ denote the set of candidate plans that satisfy executability constraints and the prescribed output schema. To improve the feasibility and verifiability of the plans, we invoke an external tool to perform evidence-based verification for each candidate and obtain plan-level structured evidence e. Based on this evidence, we construct multi-objective evidence vectors, perform Pareto-front filtering over the candidate samples, and apply a multi-agent debate–consensus mechanism to finalize the plan. The final plan selection is formalized as:

$$y^* = \mathcal{J}\big(\{(y, e) \mid y \in \mathcal{Y}\}\big), \tag{3}$$

where y^* is the finalized structured contingency plan, $\mathcal{J}(\cdot)$ denotes our evidence-constrained debate-and-consensus operator, which selects a final plan from $\mathcal{Y}$ based on the tool-based verification.

4 METHOD

Following prior work on multi-agent debate Liang et al. (2024), we adopt a **multi-agent debate** mechanism to improve plan quality and robustness: multiple LLM instances first independently

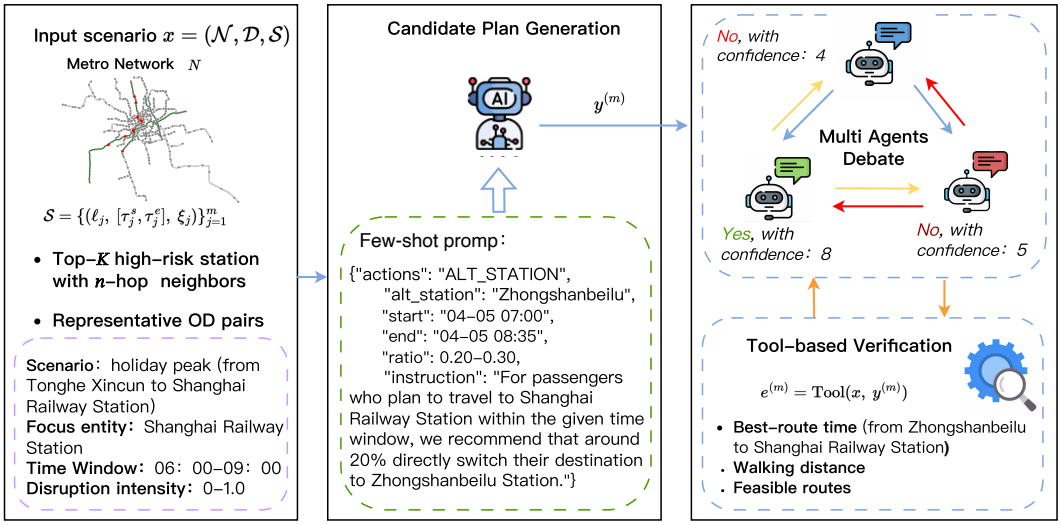

Figure 1: Overall pipeline of **MetroRehearsal**.

propose candidate plans, then engage in structured debate and cross-examination over their key disagreements, and finally converge to a finalized plan $y*$ via a consensus rule. In addition, during the debate, we incorporate an external tool for feasibility and cost verification, and use the tool outputs as evidence constraints during the debate process, which helps reduce subjective speculation and improves the interpretability and reproducibility of plan selection.

Table 1: Top-6 high demand OD pairs.

Rank	Entry	Exit	Avg. daily flow
1	Tonghe Xincun	Shanghai Railway Station	3,718
2	Jiuting	Caohejing Kaifaqu	3,635
3	Pengpu Xincun	Shanghai Railway Station	3,562
4	Xujiahui	Xinzhuang	3,453
5	Xinzhuang	Xujiahui	3,380
6	Xinzhuang	Renmin Square	3,097

4.1 Data Processing

We construct our rehearsal inputs from Shanghai metro tap-in/-out records collected over one week (April 13–19, 2015) together with the metro network topology. We first preprocess the raw data by aggregating entry and exit transactions into fixed time bins, yielding station-level inflow/outflow counts and OD demand statistics, denoted as $\mathcal{D}$. Based on these aggregates, we select a focused set of stations and time windows for rehearsal, referred to as the rehearsal subset, as shown in Tables 1 and 2, which includes:

Table 2: Top-6 high-risk stations.

Rank	Station	Avg. daily flow
1	Renmin Square	203,498
2	Shanghai Railway Station	189,410
3	Xujiahui	154,548
4	Jing'an Temple	149,360
5	Lujiazui	140,070
6	Xinzhuang	126,909

1. **Top-K high-risk station:** identified by peak intensity passenger volumes;
2. **Representative OD pairs:** OD pairs with high demand or pronounced long-tail patterns;
3. **Neighborhood spillover scope:** for each high-risk station, its n-hop neighbors (nbrs) based on $\mathcal{N}$, used to characterize potential spillover impacts;

4.2 CANDIDATE PLAN GENERATION

We constrain plan generation to a set of structured action items to ensure executability and facilitate subsequent verification. For any scenario $x = (\mathcal{N}, \mathcal{D}, \mathcal{S})$, a candidate plan is represented as

$$y = \{(a_i,\ \ell_i,\ [t_i^s, t_i^e],\ \eta_i)\}_{i=1}^{n}, \qquad a_i \in \mathcal{A}, \tag{4}$$

where a_i denotes the action type, ℓ_i is the target entity (such as station, entrance or exit), $[t_i^s, t_i^e]$ is the activation time window, and η_i is the intensity parameter. The $\mathcal{A}$ denotes the operational sets.

We generate candidates in parallel with multiple LLM instances. Given the same input x, we sample M candidate plans $\{y^{(m)}\}_{m=1}^{M}$. Each candidate is produced via template-based few-shot prompting: the prompt specifies a strict output schema (field names, data types, and valid ranges) together with a small number of demonstrations, and the model is required to output only machine-parseable structured fields. We then apply deterministic validity checks, including (i) target validity (ℓ_i must be a valid entity in $\mathcal{N}$), (ii) temporal validity ($[t_i^s, t_i^e]$ must fall within the scenario time horizon and respect the emergency windows), and (iii) parameter bounds (η_i must lie within predefined ranges). Invalid candidates are discarded or regenerated, ensuring that all plans entering the debate stage are executable.

4.3 TOOL-BASED VERIFICATION

Relying solely on internal model reasoning may lead to subjective speculation about the feasibility and cost of *diversion/alternative-station* recommendations. To ground such decisions with verifiable evidence, we introduce the Baidu Map API [1] as an external tool to validate key diversion-related items in each candidate plan and to quantify their real-world travel costs.

For each candidate plan $y^{(m)}$, we extract the subset of items that recommend *diverting to an alternative station/line*, denoted as $\mathcal{T}(y^{(m)})$. For each item in $\mathcal{T}(y^{(m)})$, we call a tool query consisting of the origin, destination, city, departure date/time window, and routing strategy parameters. The tool returns route candidates together with travel attributes such as total duration, walking distance, and transfer information.

We compress the tool responses into structured evidence and bind it to the corresponding candidate plan:

$$e^{(m)} = \mathrm{Tool}(x,\ y^{(m)}), \tag{5}$$

where $e(m)$ includes at least the following metrics, provided that there exists at least one valid route: (i) the best-route travel time $t^{(m)}$; (ii) total walking distance $w^{(m)}$; and (iii) the number of feasible route candidates $k^{(m)}$. If no feasible route is returned, we set $k^{(m)} = 0$.

Subsequently, over the candidate sample sets, we represent each candidate plan $y^{(m)}$ by a multi-objective evidence vector $\mathbf{z}^{(m)} = (t^{(m)},\ w^{(m)},\ -k^{(m)})$, and perform first-front non-dominated screening (i.e., Pareto-front extraction) (Deb et al., 2002). Concretely, a candidate plan is said to be dominated if there exists another candidate $y^{(n)}$ such that $\mathbf{z}^{(n)} \preceq \mathbf{z}^{(m)}$ and $\mathbf{z}^{(n)}$ is strictly better in at least one dimension; dominated candidates are removed. The remaining non-dominated subset constitutes the Pareto-front candidate set. This step serves as an evidence-constrained pre-screening procedure. The evidence $e^{(m)}$ is then fed into the debate stage, turning plan selection from preference-based comparison into evidence-constrained evaluation, thereby improving interpretability and reproducibility.

4.4 DEBATE AND CONSENSUS FINALIZATION

We further introduce a two-stage *debate–consensus* mechanism to finalize the candidate set and produce the final structured plan y^*. This stage takes the structured evidence returned by external tools as an explicit constraint input and requires multiple agents to conduct Debate and Consensus under the evidence constraint $e^{(m)}$. The debate stage uses structured interactions to surface potential risks such as coverage gaps, action conflicts, and overly aggressive intensities; the consensus stage then consolidates advantageous actions across candidates via voting and parameter aggregation, yielding a consistent and more conservative executable plan.

[1] https://map.baidu.com/

Debate. Given a candidate set $\{y^{(m)}\}$, we run R rounds of structured debate (we use $R = 3$). In round r, each agent produces two types of structured outputs for each candidate plan:

- **Claim**: specifies the key risk targets and corresponding time windows covered by the plan, together with its core action bundle.
- **Counter-claim**: identifies executability issues or strategic risks in other plans, with a focus on coverage gaps (e.g., missing disrupted stations or critical time windows), action conflicts, and overly aggressive intensities that may incur high operational costs (e.g., excessive inflow control or diversion at specific stations).

To quantify the credibility of each candidate under the evidence constraint and to support the subsequent consensus step, reviewer agents also assign a confidence score $c_r^{(m)} \in [1, 10]$ in each debate round. A higher $s^{(m)}$ indicates that plan $y^{(m)}$ is more credible under $e^{(m)}$. Averaging scores across rounds yields the candidate weight:

$$s^{(m)} = \frac{1}{R} \sum_{r=1}^{R} c_r^{(m)}, \tag{6}$$

Consensus. We perform consensus aggregation in two steps, action-level aligned voting and parameter summarization, and use the confidence weights from the debate stage as voting weights to produce a more conservative and executable plan under the evidence constraint.

- **Action voting**: we align identical or semantically equivalent action items (with the same $(a, \ell, [t^s, t^e])$) and compute a weighted vote total using the confidence weights of supporting candidates; actions that reach a majority threshold are retained.
- **Parameter aggregation**: for each retained action, we aggregate its intensity parameters using the median or a more conservative quantile to suppress extreme configurations and improve safety and executability.

Finally, the aggregated plans are re-checked against tool-provided evidence to ensure verifiable real-world feasibility, resulting in interpretable, reproducible, and executable final plans $\{y^*\}$.

5 EXPERIMENTS

We evaluate the quality of generated plans from three complementary dimensions. **DebateGain** is defined as the *average* improvement in the tool-evaluated score $e(m)$ achieved by the debate mechanism after it reaches consensus and converges, compared to single-shot generation. **Feasibility Rate** is defined as the proportion of feasible public-transit plans in the model's final outputs $\{y^*\}$ across different emergency scenarios, where feasibility is determined by the tool-based executability check. Finally, to characterize sensitivity to sampling randomness, we repeatedly run the method under the *Event- or holiday-induced demand surges* scenario and obtain a set of outputs $\{y^*\}$. We define **Stability** as the standard deviation of the *total walking distance* reported by the tool across these runs, a smaller value indicates more consistent and reproducible decisions under identical inputs.

5.1 RESULTS AND ANALYSIS

Table 3 compares four strategies using Deba.(↑), Feas.(↑), and Stab.(↓). Across both Qwen-3-8B Yang et al. (2025) and Llama-3-8B Dubey et al. (2024), MetroRehearsal attains the strongest overall performance, and tool verification is the key driver for boosting feasibility and improving stability.

Multi-agent debate yields quantifiable gains. MetroRehearsal consistently produces positive debate gains under both backbones, and the gains further increase when tool-based verification is enabled: *Deba.* rises from 4.21 to 15.47 on Qwen-3-8B and from 5.12 to 11.18 on Llama-3-8B. This indicates that debate-driven multi-candidate comparison can surface and correct critical deficiencies in candidate plans (e.g., insufficient station coverage or overly costly diversion choices), and tool feedback provides actionable evidence that strengthens this refinement process.

Table 3: Performance comparison across strategies with different backbones.

Strategy	Qwen-3-8B			Llama-3-8B		
	Deba.(%)↑	Feas.(%) ↑	Stab.(%)↓	Deba.(%)↑	Feas.(%) ↑	Stab.(%)↓
Single-shot	—	22.49	26.62	—	19.27	21.30
Single-shot w/ Tool Verification	—	62.34	14.76	—	54.43	15.41
MetroRehearsal w/o Tool Verification	4.21	56.97	11.08	5.12	48.04	12.96
MetroRehearsal	15.47	76.10	8.32	11.18	71.01	10.50

Tool evidence substantially improves plan feasibility. For MetroRehearsal, adding tool-based verification increases *Feas.* from 56.97% to 76.10% on Qwen-3-8B (+19.13) and from 48.04% to 71.01% on Llama-3-8B (+22.97). Moreover, under the same tool-verification setting, MetroRehearsal outperforms *Single-shot w/ Tool Verification*, improving *Feas.* by +13.76 (76.10 vs. 62.34) and reducing *Stab.* by 6.44 (8.32 vs. 14.76) on Qwen-3-8B; and improving *Feas.* by +16.58 (71.01 vs. 54.43) and reducing *Stab.* by 4.91 (10.50 vs. 15.41) on Llama-3-8B. These results suggest that the tool is not merely a post-hoc filter; instead, it provides constraint-grounded signals that guide candidate selection and consensus formation.

Debate and tool-based verification jointly improve stability. In terms of stability, MetroRehearsal achieves the lowest *Stab.* under both backbones (8.32 with Qwen-3-8B and 10.50 with Llama-3-8B), outperforming the no-tool variant (11.08 and 12.96, respectively). Notably, tool verification alone can substantially raise single-shot *Feas.* (22.49%→62.34% on Qwen-3-8B; 19.27%→54.43% on Llama-3-8B), yet the resulting plans still exhibit relatively high variability without debate (*Stab.* 14.76 and 15.41). Together, these results suggest that tool evidence provides hard constraint feedback that improves executability, whereas debate-based consensus further reduces run-to-run variance via aggregation, which makes its combination with tool evidence crucial for producing plans that are both feasible and reproducible.

5.2 Ablation Studies

Figure 2 illustrates the impact of the number of debate rounds R on the quality of rehearsal plans generated by agents built upon the Qwen-3-8B backbone. As R increases from 1 to 3, both *Debate Gain* and *Feasibility Rate* improve substantially, while *Stability* consistently decreases (lower values indicate higher stability). This trend suggests that multi-round debate effectively exposes coverage gaps and key risk targets in candidate plans, suppresses overly aggressive configurations, and thereby improves plan quality while reducing unstable action settings.

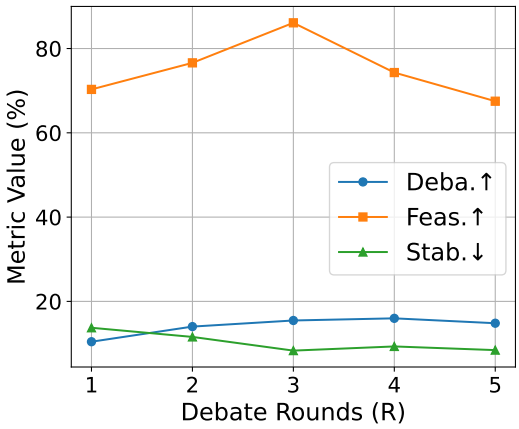

Figure 2: Ablation on the number of debate rounds R for Qwen-3-8B.

Notably, at $R = 3$, all metrics achieve an overall optimal balance: the generated plans attain strong quality gains together with the highest executability and the most stable performance. In contrast, when the number of debate rounds further increases to $R = 4$ and $R = 5$, *Debate Gain* remains marginally improved or stays at a high level, whereas the *Feasibility Rate* drops noticeably and *Stability* improvements plateau. This observation indicates that excessive debate may introduce over-correction, which in turn weakens the overall executability of the resulting plans.

6 Conclusion

We propose MetroRehearsal, a generative-agent framework for metro-hub operational emergency rehearsal. Methodologically, the framework generates structured candidate contingency plans in

parallel and incorporates a public-transit routing tool to conduct multi-round evaluations of plan feasibility and cost. It then produces a final plan through multi-agent debate and consensus convergence, enabling evidence-constrained plan selection. Experimental results show that MetroRehearsal significantly improves the feasibility of generated plans, achieves quantifiable reductions in tool-evaluated cost metrics, and produces more stable decision outputs across diverse disruption scenarios.

Future work will further expand the coverage of urban transportation tools and external evaluation modules, such as incorporating more fine-grained modeling of within-station facilities and retrieval mechanisms for intervention measures to support more detailed operational control. In addition, we plan to explore broader application scenarios and investigate the transferability and generalization of the framework across heterogeneous metro hubs in different cities.

THE USE OF LARGE LANGUAGE MODELS(LLMS)

We used a large language model (ChatGPT, GPT-4) solely for English copy-editing (grammar and style). The model was not used to design experiments, analyze data, generate results, figures, or references; all edits were reviewed by the authors, who take full responsibility for the content.

REFERENCES

Ali Asad, Stephen Obadinma, Radin Shayanfar, and Xiaodan Zhu. Reddebate: Safer responses through multi-agent red teaming debates, 2025. URL https://arxiv.org/abs/2506.11083.

Andres M Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D White, and Philippe Schwaller. Chemcrow: Augmenting large-language models with chemistry tools, 2023. URL https://arxiv.org/abs/2304.05376.

Justin Chih-Yao Chen, Swarnadeep Saha, and Mohit Bansal. Reconcile: Round-table conference improves reasoning via consensus among diverse llms, 2024. URL https://arxiv.org/abs/2309.13007.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate, 2023. URL https://arxiv.org/abs/2305.14325.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pp. arXiv–2407, 2024.

Huiying Lei, Xuedong Hua, Weijie Yu, Yongtao Zheng, and Wei Wang. Analysis of cascading failure in urban metro networks: A dynamic perspective incorporating changes in travel decisions. *Journal of Advanced Transportation*, 2025(1):5576254, 2025.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate, 2024. URL https://arxiv.org/abs/2305.19118.

Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. Chatdev: Communicative agents for software development, 2024. URL https://arxiv.org/abs/2307.07924.

Kay W Axhausen, Andreas Horni, and Kai Nagel. *The multi-agent transport simulation MATSim*. Ubiquity Press, 2016.

Yi Wang, Yuxuan Li, Xing Chen, and Yu Zhou. Urban rail transit disruption management: a decade of research on network resilience, internal, and external strategies. *Intelligent Transportation Infrastructure*, 4:liaf014, 2025.

Taylor Webb, Shanka Subhra Mondal, and Ida Momennejad. Improving planning with large language models: A modular agentic architecture, 2025. URL https://arxiv.org/abs/2310.00194.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation, 2023. URL https://arxiv.org/abs/2308.08155.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models, 2023. URL https://arxiv.org/abs/2210.03629.

Haodong Yin, Baoming Han, Dewei Li, Jianjun Wu, and Huijun Sun. Modeling and simulating passenger behavior for a station closure in a rail transit network. *PLoS one*, 11(12):e0167126, 2016.

Haodong Yin, Jianjun Wu, Zhiyuan Liu, Xin Yang, Yunchao Qu, and Huijun Sun. Optimizing the release of passenger flow guidance information in urban rail transit network via agent-based simulation. *Applied Mathematical Modelling*, 72:337–355, 2019.

Shaowei Zhang and Deyi Xiong. Debate4math: Multi-agent debate for fine-grained reasoning in math. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 16810–16824, 2025.