Enhancing Naphtha Cracking Center Scheduling via Population-Based Multi-Scenario Planning

Deunsol Yoon*, **Sunghoon Hong***, **Whiyoung Jung***, **Kanghoon Lee**, **Woohyung Lim**
LG AI Research
Seoul, South Korea
{dsyoon, sunghoon.hong, whiyoung.jung, kanghoon.lee, w.lim}@lgresearch.ai

Abstract

Naphtha Cracking Center scheduling aims to develop optimal multi-week plans under operational constraints and fluctuating demand. Our prior work (Hong et al., 2024b) introduced a multi-agent reinforcement learning (RL) system that is currently deployed in a petrochemical plant. However, standalone RL agents face several limitations: the environment is sensitive—one suboptimal action can invalidate the entire plan—and reward functions are often difficult to specify. We propose Population-Based Multi-Scenario Planning (PBMSP), a novel planning algorithm designed to complement RL agents. PBMSP maintains a diverse set of candidate schedules optimized for distinct objectives and constraints, and extends RL-based scheduling by enhancing adaptability, stability, and operational profitability.

1 Introduction

Scheduling in Naphtha Cracking Centers (NCCs) presents a fundamental challenge in petrochemical manufacturing. It involves planning a continuous, multi-stage process that converts raw naphtha into high-value products, primarily ethylene. This process includes three interdependent stages: 1) **unloading** naphtha from vessels into receipt tanks, 2) **blending** selected receipt tanks in the blending tank to achieve the target composition, and 3) **cracking** the blended feed in furnaces (LG, 2024).

The strong coupling among these stages, along with operational constraints and fluctuating demand, necessitates robust long-term scheduling. Effective plans must consider plant status, vessel arrival schedules, tank capacities, feedstock quality, and external factors such as market conditions. Based on advance shipment data, operators prepare multi-week schedules, illustrated in Figure 1, that specify which tanks receive incoming naphtha, how blending is performed, and furnace settings

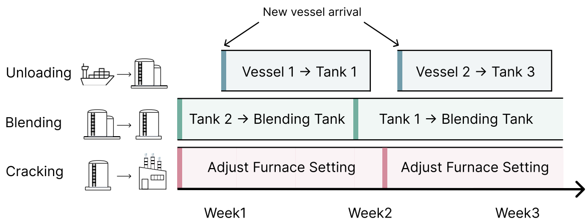

Figure 1: Simplified representation of an NCC schedule.

like feed flow rates and coil outlet temperatures. These schedules are crucial for maintaining safe, stable, and efficient operations under uncertainty.

Previous research on NCC scheduling has primarily addressed individual process components in isolation (Lee et al., 2010; Lee, 2012; Joo et al., 2023; Kim et al., 2023). Our prior work (Hong et al., 2024b)—a demonstration paper highlighting the system architecture and web-based interface[1] of its reinforcement learning (RL)-based scheduling system deployed at a petrochemical plant—proposed a cooperative multi-agent system (MAS) framework integrating the three interdependent stages of the NCC process into a unified scheduling model. In this framework, agents are assigned to manage the unloading, blending, and cracking stages respectively, and generate production plans collaboratively.

This MAS framework inherently creates operational asynchronicity, with agent actions having varied start times and durations. For example, unloading actions commence with non-periodic vessel arrivals and their durations depend on shipment volumes, while durations of blending actions vary based on receipt tank inventories. To manage it, our prior work (Hong et al., 2024b) employed the MacDec-POMDP framework (Amato et al., 2019; Xiao et al., 2022; Hong et al., 2024a; Jung et al., 2025). This framework is designed for modeling

*Equal contribution.

[1]A web-based demonstration in our earlier work is at https://www.youtube.com/watch?v=TxoWG7_SLLU.

Proceedings of the 2nd Agent AI for Scenario Planning (AgentScen), Montreal, Canada, August 16, 2025

multi-agent decision-making with asynchronicity by defining macro-actions (sequences of predefined micro-actions over multiple time steps). This representation naturally accommodates the varied start times and durations of actions inherent in the NCC.

Building on our prior work, this paper proposes a complementary planning algorithm to enhance the practical usability, robustness, and adaptability of RL-based scheduling. While our deployed multi-agent RL system demonstrates promising performance, real-world implementation reveals several challenges that limit its standalone effectiveness.

First, the NCC scheduling environment is inherently sensitive. A single suboptimal action—even one that may appear minor—can invalidate an entire schedule, eventually leading to operational failure. For instance, failing to initiate blending on time may cause receipt tank overflows, while improper blending may result in off-specification feedstock and downstream disruptions. Such fragility makes it difficult for RL agents alone to consistently produce valid and safe schedules without additional safeguards.

Second, the design of a scalar reward function for RL agents is fundamentally limited in capturing the complex, often conflicting objectives inherent to petrochemical operations. Operators must frequently balance priorities such as maximizing profitability, ensuring process stability, and satisfying operational constraints—priorities that dynamically shift based on market conditions, feedstock availability, and plant status. A static reward model cannot fully reflect these evolving trade-offs, leading to policy behaviors that may diverge from operator intent or practical feasibility.

To resolve these issues, we propose Population-Based Multi-Scenario Planning (PBMSP), a novel algorithm designed to complement existing RL agents. PBMSP maintains a diverse population of candidate schedules, each optimized under different objectives and constraint levels. This diversity enables the system to handle shifting operational criteria and priorities, providing operators with a robust set of scheduling options that better align with current plant conditions and strategic goals. Furthermore, PBMSP supports efficient asynchronous planning by identifying synchronized time points across candidate schedules, allowing fair comparisons for effective local search.

In summary, our primary contribution lies in the design and integration of PBMSP, a planning algorithm that bridges the gap between the potential of RL-based scheduling and the demands of real-world NCC operations. Through PBMSP, we enhance the usability, robustness, and adaptability of multi-agent RL systems, moving closer to their deployment in actual industrial environments.

2 Population-Based Multi-Scenario Planning (PBMSP)

This section presents our algorithm for multi-scenario scheduling, built on a structured population model. Specifically, the algorithm organizes these candidate solutions by operational characteristics and details their generation and improvement under varying constraint levels.

2.1 Structured Population Design

Unlike approaches that rely on a single population (Jaderberg et al., 2017; Jung et al., 2020; Parker-Holder et al., 2020; Wu et al., 2023; Zhao et al., 2023), our framework structures the population into distinct groups.

Each group is associated with a specific operational scenario. This scenario is defined by a unique combination of an operational criterion and an operating level. The criterion is evaluated by a scalar fitness function reflecting aspects like profitability and stability. The operating level dictates the stringency of operational constraints. These levels span a spectrum from conservative (using a limited control range) to stressed (pushing equipment operation to near its critical limits). A higher level signifies more restrictive operational constraints.

The resulting hierarchical structure of operating levels presents a useful characteristic. Schedules satisfying stricter constraints (higher level) inherently meet looser ones (lower level), potentially enabling the transfer of promising solutions across different operational priorities. This design choice mirrors real-world NCC operations where constraint stringency naturally varies based on plant conditions or goals; for instance, stressed levels might suit high demand while conservative levels prioritize safety during stable periods.

By adopting this structured population, we aim to leverage the inherent benefits of population-based approaches—such as parallel exploration and local optima escape—while directly addressing the challenges posed by the multi-faceted nature of NCC scheduling. The dedication of specific groups to distinct scenarios is intended to improve the efficiency and effectiveness of the search process.

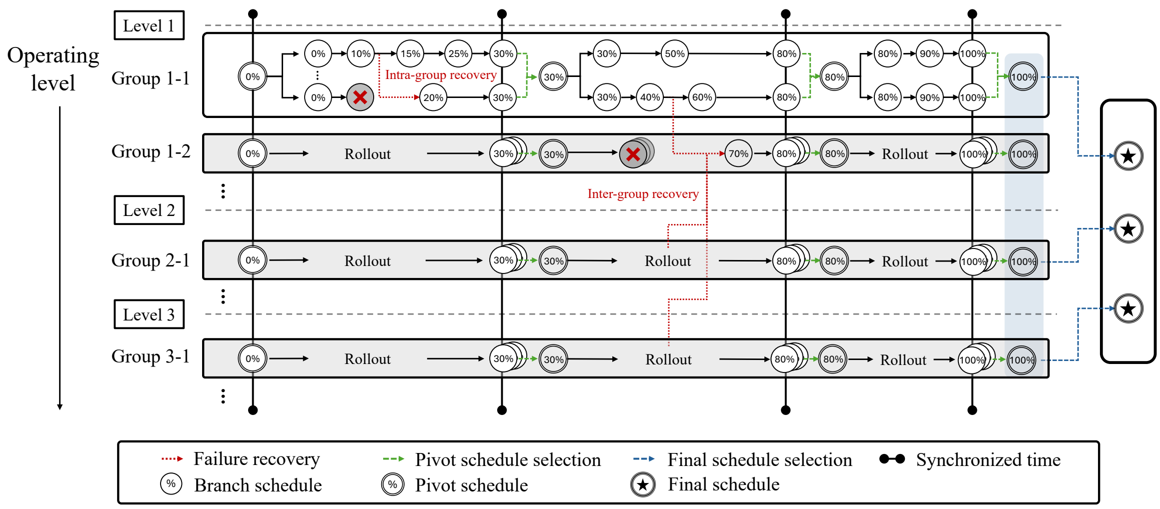

Figure 2: Multiple groups, each tied to a specific operational criterion and operating level, maintain branch schedules progressing via iterative rollouts (completion percentage tracked). At synchronized times, each group's pivot schedule is chosen from its branch schedules and those of equal or higher level groups. Failures are managed by intra/inter-group recovery. The final schedules are selected from the set of complete pivot schedules. For clarity, Group 1-1's rollout is detailed; others are brief.

This organization facilitates targeted exploration under diverse operational priorities and constraints, with the expectation of yielding a more comprehensive and robust set of high-quality schedules compared to a uniformly explored population.

2.2 Schedule Construction Process

Based on the group defined above, our framework generates a diverse set of schedules through the following iterative process as depicted in Figure 2.

Initialization At the beginning of the planning, each group's pivot schedule—which serves as the current best-known solution and the baseline for exploration for that group's scenario—is initialized as an empty sequence of macro-actions, reflecting the initial status of the NCC system.

Iterative construction The following steps are repeated iteratively until planning horizon:

(1) Pivot-based branching: The pivot schedule is replicated in parallel to create multiple branch schedules. This allows a broad exploration of various alternative decisions.

(2) Rollout based on scenario: Each branch schedule undergoes a rollout process considering its associated group's scenario. This process progressively constructs a complete schedule by sequentially applying macro-actions.

(3) Synchronized evaluation and update: At predefined *synchronized time*—specific moments where all branch schedules have reached an identical operational time (e.g., a shared event like a vessel arrival)—each group evaluates not only its own branch schedules but also those from all other groups operating at an equal or higher level, based on their respective fitness functions. This strategy facilitates the discovery of solutions that effectively balance diverse priorities and constraints.

Throughout the rollout process, the algorithm incorporates a robust failure recovery mechanism.

- *Intra-group recovery*: A failing schedule within a group is replaced by a copy of the current best-performing schedule in that group (based on its fitness), and rollout continues.

- *Inter-group recovery*: If all schedules in a group fail, the entire group is re-initialized with a copy of the best-performing schedule from all groups at an equal or higher operating level, and rollout resumes.

Furthermore, if all schedules across all groups fail, the algorithm restarts exploration from each group's pivot schedule at the last synchronized time. These layered mechanisms enhance the planning robustness by preventing premature termination.

Final schedule selection Once the iterative planning process is complete, a final selection step is performed. Instead of directly presenting all group-specific pivot schedules, a separate set of final evaluation criteria is applied to assess these complete schedules. This is because, unlike the fitness functions used during the planning process, the final criteria can consider aspects that can only be accu-

Methods	Success Rate (%)	Normalized Return	Time (min.)
PBMSP (Full resources for parallel rollout)	93.8	0.994	38.8
PBMSP (50% resources for parallel rollout)	87.5	0.988	37.8
Simple RL Rollout (10k sampling)	37.5	0.922	516.5
Simple RL Rollout (1k sampling)	12.5	0.926	57.1

Table 1: Quantitative comparison of PBMSP and Simple RL Rollout methods.

rately assessed once a complete schedule has been generated. The top-performing schedules, according to these final criteria, are then presented to the human operator for review and implementation.

3 Evaluation

Quantitative Analysis We assess the effectiveness of the proposed method through experiments based on diverse expert-designed backtest schedules. Each schedule captures the full information describing the operational status of the NCC plant at the time of scheduling, including inventory levels, equipment availability, and process constraints.

Due to confidentiality agreements with industry collaborators, we omit specific configuration details and parameter values; however, results are presented in an abstracted form that faithfully reflects the comparative performance and key insights.

We compare two methods for schedule generation based on a pre-trained RL policy from our prior work (Hong et al., 2024b): 1) **PBMSP**, our proposed method that actively explores diverse schedule groups, and 2) **Simple RL Rollout**, a baseline that samples 1,000 or 10,000 schedules using the policy and selects the highest-return one that successfully completes. All experiments were performed on two AMD EPYC 7453 28-core processors, with both methods parallelized to fully utilize available resources.

We evaluate these methods using key metrics:

- *Success Rate*: The average success rate in generating a complete schedule without failure.

- *Normalized Return*: The maximum return among successfully generated schedules. Normalized by the maximum return across all methods for each specific data point.

- *Wall-clock Time*: The average time taken to generate a schedule across all data points.

PBMSP consistently outperforms Simple RL Rollout by generating schedules from a broader range of initial operational status (higher success rate) and achieving higher returns, while also requiring significantly less time due to more efficient sampling. Although PBMSP (50% resources) has similar wall-clock time due to parallelization, its reduced group size leads to fewer schedules and thus lower performance than the full PBMSP.

Qualitative Insights from Deployment We introduced an online web service (Hong et al., 2024b) to optimize NCC operational schedules. This platform enabled users to upload the current operational status and generate schedules. The service then presented these schedules with figures and statistics in staff-friendly downloadable formats.

We have integrated PBMSP into this web service. Feedback from operators indicates this integration has significantly enhanced the service's real-world utility, delivering several key improvements:

- *Enhanced Schedule Generation and Utilization*: The frequency of generating successful schedules has dramatically increased. This allows users to rely on service-generated schedules more often in practice, leading to greater operational reliability and reduced need for manual intervention.

- *Diverse and Adaptable Schedule Offerings*: The service now provides a broader range of successful schedules. This variety gives users the flexibility to select schedules that best align with their current operational priorities.

- *Increased Profitability*: Backtesting data reveals that the PBMSP-enhanced service consistently generates more profitable schedules compared to those created by human experts.

4 Conclusion

This paper presents PBMSP, a population-based approach that enhances NCC scheduling to overcome the limitations of standalone RL. By maintaining a diverse population of candidate schedules optimized for varied objectives and constraints, it improves schedule completeness and efficiency, as confirmed by operator feedback. PBMSP also shows strong potential for broader industrial optimization problems with dynamic constraints and can contribute to the planning capabilities increasingly needed by modern large language models.

References

Christopher Amato, George Konidaris, Leslie P Kaelbling, and Jonathan P How. 2019. Modeling and planning with macro-actions in decentralized POMDPs. Journal of Artificial Intelligence Research, 64:817–859.

Sunghoon Hong, Whiyoung Jung, Deunsol Yoon, Kanghoon Lee, and Woohyung Lim. 2024a. Agent-oriented centralized critic for asynchronous multi-agent reinforcement learning. In International Conference on Autonomous Agents and Multiagent Systems Workshop on Adaptive and Learning Agents.

Sunghoon Hong, Deunsol Yoon, Whiyoung Jung, Jinsang Lee, Hyundam Yoo, Jiwon Ham, Suhyun Jung, Chanwoo Moon, Yeontae Jung, Kanghoon Lee, Woohyung Lim, Somin Jeon, Myounggu Lee, Sohui Hong, Jaesang Lee, Hangyoul Jang, Changhyun Kwak, Jeonghyeon Park, Changhoon Kang, and Jungki Kim. 2024b. Naphtha cracking center scheduling optimization using multi-agent reinforcement learning. In International Conference on Autonomous Agents and Multiagent Systems.

Max Jaderberg, Valentin Dalibard, Simon Osindero, Wojciech M. Czarnecki, Jeff Donahue, Ali Razavi, Oriol Vinyals, Tim Green, Iain Dunning, Karen Simonyan, Chrisantha Fernando, and Koray Kavukcuoglu. 2017. Population based training of neural networks. arXiv preprint arXiv:1711.09846.

Chonghyo Joo, Hyukwon Kwon, Junghwan Kim, Hyungtae Cho, and Jaewon Lee. 2023. Machine-learning-based optimization of operating conditions of naphtha cracking furnace to maximize plant profit. Computer Aided Chemical Engineering, 52:1397–1402.

Whiyoung Jung, Sunghoon Hong, Deunsol Yoon, Kanghoon Lee, and Woohyung Lim. 2025. Agent-centric actor-critic for asynchronous multi-agent reinforcement learning. In International Conference on Machine Learning.

Whiyoung Jung, Giseung Park, and Youngchul Sung. 2020. Population-guided parallel policy search for reinforcement learning. In International Conference on Learning Representations.

Jeongdong Kim, Chonghyo Joo, Minsu Kim, Nahyeon An, Hyungtae Cho, Il Moon, and Junghwan Kim. 2023. Multi-objective robust optimization of profit for a naphtha cracking furnace considering uncertainties in the feed composition. Expert Systems with Applications, 216:119464.

Ho-kyung Lee. 2012. Method for naphtha storage tank operation and system for the same. KR Patent App. KR1020100046570A.

Taeyeong Lee, Jun-Hyung Ryu, Ho-Kyung Lee, and In-Beum Lee. 2010. A study on scheduling of naphtha transportation and storage systems for naphtha cracking center. Chemical Engineering Research and Design, 88(2):189–196.

LG. 2024. What happens when a 24-hour AI system is introduced to petrochemical plant? https://www.youtube.com/watch?v=UBlgcgluIjU.

Jack Parker-Holder, Aldo Pacchiano, Krzysztof M Choromanski, and Stephen J. Roberts. 2020. Effective diversity in population based reinforcement learning. In Advances in Neural Information Processing Systems, volume 33, pages 18050–18062.

Shuang Wu, Jian Yao, Haobo Fu, Ye Tian, Chao Qian, Yaodong Yang, Qiang Fu, and Yang Wei. 2023. Quality-similar diversity via population based reinforcement learning. In International Conference on Learning Representations.

Yuchen Xiao, Weihao Tan, and Christopher Amato. 2022. Asynchronous actor-critic for multi-agent reinforcement learning. In Advances in Neural Information Processing Systems, volume 35, pages 4385–4400.

Rui Zhao, Jinming Song, Yufeng Yuan, Haifeng Hu, Yang Gao, Yi Wu, Zhongqian Sun, and Wei Yang. 2023. Maximum entropy population-based training for zero-shot human-AI coordination. In The AAAI Conference on Artificial Intelligence, pages 6145–6153.