Multi-Agent LLMs for Automating Sustainable Operational Decision-Making

Emma Pajak[a], Abdullah Bahamdan[a], Klaus Hellgardt[a], Antonio del Río-Chanona[a,*]

[a] Imperial College London, Department of Chemical Engineering, London, United Kingdom.
* Corresponding Author: a.del-rio-chanona@imperial.ac.uk.

ABSTRACT

Operational decision-making in Process Systems Engineering (PSE) has achieved high proficiency at specific levels, such as supply chain optimization and unit-operation optimization. However, a critical challenge remains: integrating these layers of optimization into a cohesive, hierarchical decision-making framework that enables sustainable and automated operations. Addressing this challenge requires systems capable of coordinating multi-level decisions while maintaining interpretability and adaptability. Multi-agent frameworks based on Large Language Models (LLMs) have demonstrated significant promise in other domains, successfully simulating traditional human decision-making tasks and tackling complex, multi-stage problems. This paper explores their potential application within operational decision-making for PSE, focusing on sustainability-driven objectives. A realistic Gas-Oil Separation Plant (GOSP) network is used as a case study, mimicking a hierarchical workflow that spans from initial back-of-the-envelope multi-objective optimization for cost-emissions trade-offs to a negotiation phase reflecting upper management decision-making, and culminating in high-fidelity simulations to validate operational setpoints at the plant level. This workflow serves as a canvas to assess the benefits of multi-agent LLMs, including their ability to integrate multi-layered decisions, enhance the explainability of strategies, and streamline automation in PSE workflows. The results demonstrate the potential of multi-agent LLMs to address the integration challenge in PSE, supporting sustainable and efficient operational decisions. Beyond GOSPs, this research highlights promising applications of multi-agent LLMs across process engineering, contributing to the vision of hierarchical, automated decision-making for the 'plant of the future,' where diverse models and tools operate within an intelligent, unified framework.

Keywords: operational decision-making, large language models (LLMs), sustainability, optimization.

INTRODUCTION

Operational decision-making (ODM) is pivotal in Process Systems Engineering (PSE), serving as the foundation for optimizing the performance, efficiency, and sustainability of industrial processes. In an era where industries face mounting pressure to achieve net zero emissions while maintaining cost-effectiveness, the necessity for robust ODM frameworks has become increasingly critical. Optimization in PSE is conducted at multiple hierarchical levels: system-wide optimization enhances supply chain efficiency by managing the flow of materials, information, and funds; plant-wide optimization focuses on improving operational efficiency within individual plants, such as utility management; and unit operation optimization targets specific processes like reactor performance. Despite these layered approaches, a significant challenge remains—the automation and seamless integration of multi-layered decision-making processes. As industrial systems grow in complexity, there is a need for automated tools that can unify these layers, providing operators with rapid access to comprehensive insights or suggestions. Such automation not only promises substantial cost savings but also facilitates significant reductions in emissions, thereby driving industries toward more sustainable and economically viable operations.

An increasing number of studies are emerging that focus on leveraging Generative Artificial Intelligence (GenAI) within PSE [1]]. GenAI encompasses a range of technologies, including Large Language Models (LLMs),

which are advanced AI systems trained on vast amounts of text data to understand and generate human language. LLMs have found applications in PSE such as supporting the automatic generation of P&IDs [2], and fault detection and diagnosis in chemical processes [3].

Multi-Agent LLMs

Multi-agent LLMs are a development in GenAI that extends the capabilities of individual language models by orchestrating multiple specialized LLMs, known as agents, to collaborate on complex tasks [4]. Studies indicate that multi-agent LLM systems have the potential to surpass single-agent approaches in both accuracy and complex problem-solving capabilities [5]. In tasks replicating human behavior, multi-agent architectures have demonstrated substantial accuracy gains, with some findings suggesting a direct correlation between the number of agents and overall performance [5]. The "More Agents Is All You Need" approach (Agent Forest) further shows that increasing the number of instantiated LLM agents through sampling and voting can boost accuracy, particularly on difficult tasks [6]. Meanwhile, the Mixture-of-Agents (MoA) approach leverages iterative collaboration among multiple models to address more intricate problems, outperforming single agents on benchmarks such as AlpacaEval 2.0, MT-Bench, and FLASK [7].

Building on these performance advantages, multi-agent LLMs excel in advanced multi-step problems through agent specialization: specialized agents can be developed and dedicated to distinct subtasks or areas of expertise, avoiding reliance on a single model for all problem aspects. Hence, multi-agent systems are inherently modular and thus flexible, making them easier to reconfigure as operational needs evolve. Adding or updating agents—each responsible for a particular function—can often be done with minimal disruption to the broader system. This adaptability ensures that multi-agent LLM frameworks remain robust and scalable, even in rapidly changing industrial or research environments.

Multi-agent LLMs have already seen application across a wide range of industries [8]. For example, ChatDev integrates different phases of the software development lifecycle into a unified communication system using LLM agents. It enhances collaboration through multi-turn dialogues, chat chain organization, and dehallucination, improving software design, coding, and debugging [9]. Compared to its single-LLM counterpart, GPT-Engineer, ChatDev achieves higher Quality (0.3953 vs. 0.1419) and Consistency (0.8021 vs. 0.7887). These results demonstrate that multi-agent systems can outperform single-agent systems, particularly in managing complex, multi-stage tasks [9].

More recent works have demonstrated the implementation of multi-agent systems in process engineering contexts. For example, industrial automation of control systems has been explored using multi-agent frameworks with LLM-based agents, incorporating validation and reprompting architectures to improve accuracy, adapt to dynamic disturbances, and enhance reliability in complex operational environments [10].

Study Aim

Although multi-agent LLMs have demonstrated success across various industries and are beginning to gain traction in PSE, there remains a substantial opportunity to harness their capabilities specifically for ODM. By leveraging their capacity for complex problem-solving and coordination of multi-step tasks through specialized agents, these systems could serve as a decision-support tool that fully automates and integrates ODM. This, in turn, unlocks significant cost savings and emission reductions, supporting PSE operations in meeting global sustainability targets. This work investigates the application of multi-agent LLM systems for ODM in PSE. This study explores how these systems can represent multiple stakeholders across the decision-making landscape to address the complexities of hierarchical workflows, and focuses on the following key areas:

- **Integration:** Assessing how multi-agent LLMs can integrate the different layers of decision-making, from management to operator level.
- **Negotiation:** Investigating the potential of agents to simulate realistic stakeholder negotiations, addressing trade-offs between competing business units and goals such as economic performance and sustainability.
- **Automation:** Exploring how multi-agent LLMs can reduce human intervention while maintaining transparency and explainability, ensuring that the system's decisions are interpretable and actionable.

Together, these focal points aim to assess the feasibility and transformative potential of multi-agent LLMs in enabling sustainable, efficient, and intelligent decision-making within PSE.

METHODOLOGY

To explore the potential of multi-agent LLM systems for operational decision-making, it is essential to select a case study that is both realistic and representative of industrial challenges. This curated case study draws inspiration from production planning in the oil and gas industry, a sector characterised by extensive, interconnected supply chains and complex, interdependent decision-making processes that are needed of sustainable operations. The detailed case study, optimization formulation, and HYSYS flowsheet can be found in the work by Bahamdan et al. [11]. This example not only captures multi-layered workflows typical of PSE but also integrates simulation and optimisation components, making it an ideal

testbed for assessing LLMs' tool usage. Furthermore, the cost-emissions trade-off provides a compelling scenario to explore LLM agent negotiation, mirroring real-world industrial decision-making dynamics.
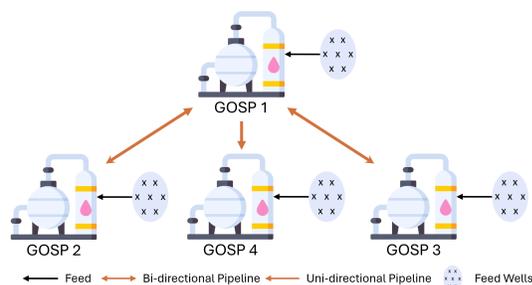
Gas-Oil Separation Plant (GOSP) Case Study

Figure 1. Schematic of four interconnected GOSPs connected by uni- and bi-directional pipelines.

This study focuses on the midstream sector of the oil and gas supply chain, which forms the logistical backbone, facilitating the production, processing, and transfer of products. This sector comprises a network of Gas-Oil Separation Plants (GOSPs) that separate oil, gas, and water phases, remove impurities, and ensure products meet customer specifications. These facilities are interconnected via swing pipelines to optimize operations and handle disturbances effectively. A simplified network of four GOSPs, as illustrated in Figure 1, is considered for this study. Each GOSP processes feedstock from a dedicated cluster of production wells. Although all wells within a cluster produce oil of the same grade, feed quality varies slightly but remains within acceptable limits.

Typical GOSPs consist of three main sections: crude oil, associated gas, and oily water. Each incorporates a series of separation vessels and energy-intensive rotating equipment, which contribute significantly to emissions in fossil-based fuel production. This case study builds on recent work that employs state-of-the-art, data-driven methods to decarbonize and optimize GOSP networks by utilizing swing pipelines.

Hierarchical Decision-Making Workflow

Building on the previously defined 4 GOSP system, this case study examines the production planning process of a typical oil and gas company involving multiple stakeholders from different organizations. Within industry, the production planning process typically follows the following structure. A month in advance, a central planning group receives government-issued monthly production targets. An upstream team uses advanced reservoir simulations to determine optimal production capacities for each well, coordinating with midstream planners to ensure processing systems are ready. Once all parties agree on production targets and midstream capacities,
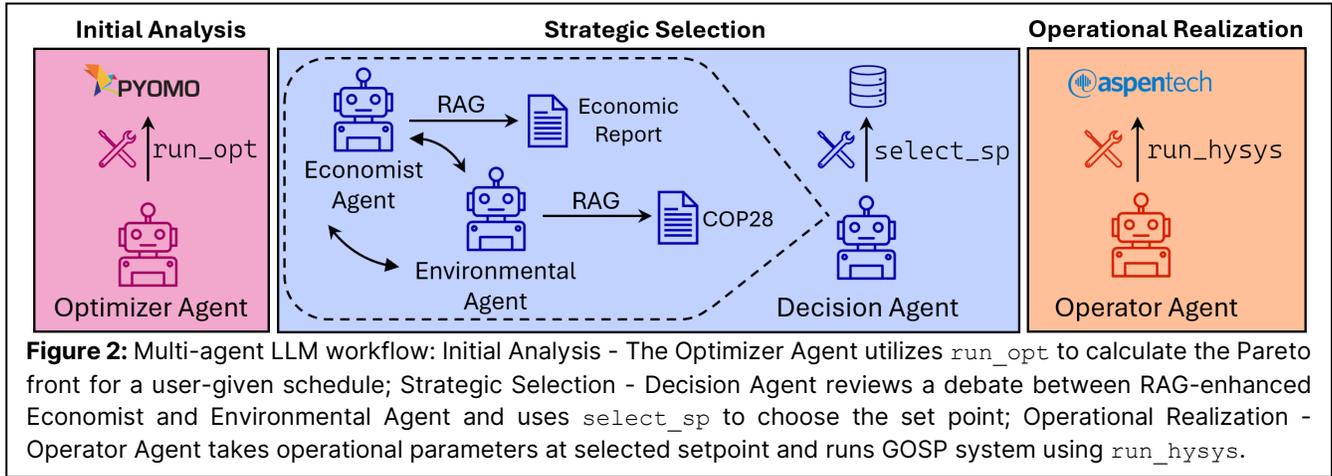
the decision is finalized. Afterwards, separate organizations for production and environmental compliance monitor real-time data to make necessary adjustments for the remainder of the planning period.

Closely mirroring the status quo of industry, this study's production planning process takes the following structure. The objective of the ODM workflow is to determine an operational setpoint for a given schedule, defining the system's cost and emissions through a three-stage decision-making process. The '**Initial Analysis**' stage employs Multi-Objective Optimization (MOO) to balance cost and emissions, generating a Pareto front of operational set points. As Operating Expenses (OPEX) are primarily driven by the power consumption of major equipment, a low-fidelity surrogate was trained on plant operation data to avoid expensive, iterative runs of a process simulation. The surrogate outputs the power consumption of the four major rotating equipment that represent 90% of the total power consumption, which is used as a proxy for cost. Emissions, including scope 1 (flaring and venting) and scope 2 (indirect emissions from power generation) are calculated by using mass balances to determine surplus gas. Model constraints include design capacities for the GOSPs and swing pipelines, material balances, and flow direction of swing pipelines. This problem is formulated as a Mixed-Integer Linear Program (MILP), determining the optimal feed rate for each facility, as well as the swing pipe configuration. The '**Strategic Selection**' involves choosing an operational setpoint from the Pareto front by evaluating trade-offs between cost and emissions, aligning with the management's strategic priorities and considering market and regulatory factors. Finally, the '**Operational Realization**' stage implements the chosen setpoint in system operations, transitioning from theoretical optimization to practical application while accounting for operational uncertainties such as equipment performance variations and external disruptions.

Multi-Agent LLM Framework

LLMs and Architecture

The workflow employs two LLM models, both deployed locally using Ollama on an RTX 3060 GPU. Llama-3.1 8B is used for reasoning and decision-making tasks, such as debate evaluation and argument generation. For tasks requiring tool integration, the Llama-3-Groq-8B-Tool-Use model is used, which is fine-tuned for structured tool use and function calling. This model scores 89% overall accuracy on the Berkeley Function Calling Leaderboard used for benchmarking LLM tool usage [12]. The agentic system was implemented using LangGraph, selected for its ability to build workflows around a graph structure. In this framework, agents are represented as nodes, and edges define the routing logic, making the workflow intuitive and easy to visualise. Multi-agent LLM

Figure 2: Multi-agent LLM workflow: Initial Analysis - The Optimizer Agent utilizes `run_opt` to calculate the Pareto front for a user-given schedule; Strategic Selection - Decision Agent reviews a debate between RAG-enhanced Economist and Environmental Agent and uses `select_sp` to choose the set point; Operational Realization - Operator Agent takes operational parameters at selected setpoint and runs GOSP system using `run_hysys`.

architectures refer to the connections, or edges, that direct workflows between agents [13]. Different architectures are suited to different tasks. For example, sequential architectures are simple and efficient, making them ideal for workflows with clearly defined stages. This workflow employs an architecture that combines the sequential structure to address each decision-making layer systematically and incorporates a cycle for the setpoint debate between Environmental and Economist agents. This design enables focused negotiation, improving decision quality through iterative exchanges without unnecessary complexity.

Agents and Tools

The workflow relies on three specialized tools to facilitate decision-making and system simulation, as summarised in Table 1.

Table 1: Workflow tools and their functionalities.

Tool	Description
`run_opt`	Takes the initial feed, executes Pyomo-based MOO, and returns Pareto front.
`choose_sp`	Takes setpoint identifier, retrieves the Pareto front, and returns the selected operational parameters.
`run_hysys`	Takes finalized operating schedule, runs a HYSYS simulation, returns key unit operations power consumption and OPEX.

The framework also integrates Retrieval-Augmented Generation (RAG), which enhances LLM responses by incorporating external documents or knowledge bases, enabling a more informed and contextually relevant output [14]. Both the Environmental and Economic agents employed RAG to strengthen their arguments. The Environmental agent referenced sustainability-focused documents, such as the COP28 report [15]. Meanwhile, the Economic agent had access to economic and corporate documents to support its arguments.

Table 2: Overview of agent roles in the workflow.

Agent	Role Description
Optimisation Agent	Extracts the ICF from user input and runs the `run_opt` tool to compute the Pareto front, balancing cost and emissions.
Economic Agent	Advocates for cost minimisation in the debate phase.
Environmental Agent	Advocates for emissions reduction, using RAG on documents such as COP28 in the debate phase.
Decision Agent	Evaluates arguments, selects the setpoint, and retrieves operational parameters with `choose_sp` tool.
Operator Agent	Uses `run_hysys` to run operations and validate cost and emissions.

Agentic Workflow Description

Figure 2 illustrates the agentic workflow aligned with the three-stage decision-making process. It begins with the Optimisation Agent, which extracts the Initial Condition Factors (ICF) from the user prompt and uses its `run_opt` tool to execute a Pyomo-based MOO. This generates a set of feasible operational configurations for the GOSP which forms a Pareto front representing the trade-off between cost and emissions.

The Economic and Environmental Agents then engage in a structured debate to select the operating setpoint. The Economic Agent advocates for cost minimization, while the Environmental Agent prioritises emissions reduction, using RAG with sources like COP28 to support its arguments. Each agent presents two arguments in turn, advocating for their preferred setpoint.

Next, the Decision Agent evaluates the debate transcript to select the winning setpoint based on argument quality. Using the `choose_sp` tool, it retrieves the corresponding operational parameters from the Pareto front. The Operator Agent uses these parameters with the `run_hysys` tool to interact with the HYSYS simulation,

set stream values, and retrieve power consumption data to calculate the actual OPEX for the chosen configuration.

RESULTS

The multi-agent LLM framework successfully executed the entire operational workflow in all trials. Upon receiving a user prompt with the latest ICF schedule, the Optimizer Agent generated the Pareto front using the `run_opt` tool. Subsequently, the Decision Agent evaluated the arguments presented by the Economist and Environmentalist agents to select the optimal operational setpoint. Finally, the Operator Agent applied the parameters from the chosen configuration to operate the GOSP network. Across ten runs, the agentic workflow had an average runtime of 79.2 seconds on an RTX 3060 GPU.

Agent Debate and Decision-Making Process

To evaluate the multi-agent LLM framework's responsiveness to evolving information, two distinct runs were conducted under identical ICF schedules with differing informational inputs. From the back-and-forth exchange between the agents, two quotes are presented to highlight some of the more significant arguments.
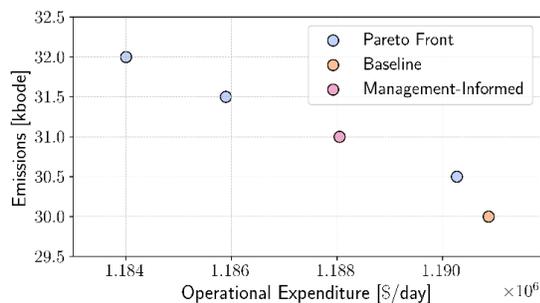

Figure 3. Pareto front illustrating the trade-off between operational costs and emissions. Each point represents a feasible operational configuration for the GOSP network.

Baseline Run

In the initial run, no additional economic directives were provided. The agents engaged in a debate focused solely on balancing emissions and operational costs. The Environmentalist Agent cited global sustainability commitments, emphasizing recent international agreements:

```
    At COP28, Parties emphasized the im-
portance of accelerating emissions reductions
and promoting a just transition for energy
systems worldwide. If we fail to make ambi-
tious choices now, we risk undermining these
global commitments.        -Environmental Agent
```

Influenced by these arguments, the Decision Agent opted for a setpoint that minimised emissions, resulting in the highest operational cost among available options.

Second Run (With Economic Directive)

In the next run, an "Economic Directive" document was introduced to the Economist Agent via RAG. This directive stipulated a soft constraint on operational costs of 1,188,500 [\$/day], from upper management. Equipped with this information, the Economist Agent adjusted its arguments to align with the new budgetary constraint:

```
Given upper management's directive to keep
costs below $1,188,500 per day, we must ex-
plore a more balanced trade-off. Setpoint 3
adheres to the strict budget while mitigating
emissions.              -Economist Agent
```

Consequently, the Decision Agent adopted a setpoint that achieved a balanced compromise between emissions and costs, ensuring compliance with the managerial cost threshold while still addressing environmental concerns. This outcome marked a clear departure from the earlier sustainability-focused solution, illustrating how new information can prompt a recalibration of priorities in the workflow, resulting in a shift in outcome.

Overall, these findings highlight the adaptability of the multi-agent LLM framework. When presented with evolving objectives—be it a sustainability pledge linked to COP28 or a budgetary mandate from upper management—the agents' internal debate naturally reorients toward fulfilling the most pressing constraint. In doing so, the system demonstrates how LLMs, when integrated into an orchestrated multi-agent workflow, can reconcile environmental imperatives with operational costs, integrating new information into decision-making processes. A workflow like the one presented in this work could integrate into the production planning cycle, enabling real-time adjustments to operational parameters to optimize the alignment between planned and actual performance.

CONCLUSION

In line with this study's aim to explore the potential of multi-agent LLMs for ODM in PSE, our findings demonstrate the efficacy of these frameworks within the midstream sector of the oil and gas supply chain. By leveraging multi-agent LLMs, the framework effectively integrates complex, multi-level decision-making processes in an automated and highly interpretable manner, dynamically adapting to evolving operational constraints and directives. This highlights the ability of multi-agent LLMs to navigate trade-offs, balancing sustainability and economic objectives.

Presented as a decision-support tool to complement human expertise, the multi-agent LLM framework accelerates stakeholder alignment by providing a structured and interpretable approach to complex decision-making. In real deployments, domain experts will remain vital to validating final setpoints, and ensuring adherence to plant safety protocols, corporate objectives, and

operational constraints. Although multi-agent LLMs are rapidly advancing, they still face challenges, such as scalability, model alignment, and governance.

Future Work

One key direction is expanding argument quality and industrial relevance through RAG by integrating dynamically updated reference documents (e.g., industry standards, real-time news) and proprietary materials (e.g., sustainability reports, financial data). Incorporating live economic or environmental metrics would enhance both precision and relevance. Another focus is generalization and scalability: by adjusting the `run_opt` tool and the MILP, the workflow could handle different numbers of GOSPs of varying capacities. To further validate the benefits of multi-agent LLMs for ODM in PSE, work should explore their use in other sectors, e.g., chemical manufacturing.

Introducing additional agent roles (e.g., Regulatory Agent, Safety Manager) and enabling agents to learn from prior runs or user feedback would enrich the decision-making process. User interaction and overrides also warrant attention, allowing domain experts to review and override decisions, supported by access to Pareto fronts, and debate transcripts. Finally, assessing the framework's robustness under incomplete or conflicting information is critical to ensure feasible outcomes. Addressing ethical and interpretability concerns is equally important for building trust, requiring alignment with ethical standards and clear explanations of agent reasoning.

ACKNOWLEDGEMENTS

REFERENCES

1. Decardi-Nelson B, Alshehri AS, Ajagekar A, You F, Generative AI and process systems engineering: The next frontier. *Comput Chem Eng*, 187:108723 (2024) doi: 10.1016/j.compchemeng.2024.108723.
2. Hirtreiter E, Balhorn LS, Schweidtmann AM, Towards automatic generation of Piping and Instrumentation Diagrams (P&IDs) with Artificial Intelligence. *AIChE J* 70: e18259 (2024), doi: 10.1002/aic.18259.
3. Khan A, Nahar R, Chen H, Flores GEC, Li C, FaultExplainer: Leveraging Large Language Models for Interpretable Fault Detection and Diagnosis. Dec. 19, 2024, *arXiv*: doi: 10.48550/arXiv.2412.14492.
4. Guo T *et al*. Large Language Model based Multi-Agents: A Survey of Progress and Challenges. Apr. 19, 2024, *arXiv*: doi: 10.48550/arXiv.2402.01680.
5. Sreedhar K, Chilton L. Simulating Human Strategic Behavior: Comparing Single and Multi-agent LLMs. Jul. 02, 2024, *arXiv*: doi: 10.48550/arXiv.2402.08189.
6. J. Li, Q. Zhang, Y. Yu, Q. Fu, and D. Ye, "More Agents Is All You Need," Oct. 11, 2024, arXiv: doi: 10.48550/arXiv.2402.05120.
7. J. Wang, J. Wang, B. Athiwaratkun, C. Zhang, and J. Zou, "Mixture-of-Agents Enhances Large Language Model Capabilities," Jun. 07, 2024, arXiv: doi: 10.48550/arXiv.2406.04692.
8. X. Li, S. Wang, S. Zeng, Y. Wu, and Y. Yang, "A survey on LLM-based multi-agent systems: workflow, infrastructure, and challenges," Vicinagearth, vol. 1, no. 1, p. 9, Oct. 2024, doi: 10.1007/s44336-024-00009-2.
9. C. Qian et al., "ChatDev: Communicative Agents for Software Development," in Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), L.-W. Ku, A. Martins, and V. Srikumar, Eds., Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 15174–15186. https://aclanthology.org/2024.acl-long.810
10. [J. Vyas and M. Mercangöz, "Autonomous Industrial Control using an Agentic Framework with Large Language Models," Nov. 08, 2024, arXiv: doi: 10.48550/arXiv.2411.05904.
11. Bahamdan A, Shah N, del Rio-Chanona A. Surrogate Based Mixed Integer Linear Programming Model for Decarbonization of an Integrated Gas-Oil Separation Network. *Comput Aided Chem Eng*, 53:2209–2214 (2024). doi: 10.1016/B978-0-443-28824-1.50369-0.
12. Hugging Face Llama-3-8B. https://huggingface.co/Groq/Llama-3-Groq-8B Tool-Use.
13. T. Händler, "Balancing Autonomy and Alignment: A Multi-Dimensional Taxonomy for Autonomous LLM-powered Multi-Agent Architectures," Oct. 05, 2023, arXiv: doi: 10.48550/arXiv.2310.03659.
14. P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in Proc 34th Int Conf on Neural Information Processing Systems, in NIPS '20. Red Hook, NY, USA: Curran Associates Inc., Dec. 2020, pp. 9459–9474.
15. "Summary of Global Climate Action at COP 28 │ UNFCCC." https://unfccc.int/documents/636485.