Proceedings of the 2025 Winter Simulation Conference
E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.

ADVANCING MILITARY DECISION SUPPORT: REINFORCEMENT LEARNING-DRIVEN SIMULATION FOR ROBUST OPERATIONAL PLAN VALIDATION

Michael Möbius[1], Daniel Kallfass[2], Stefan Göricke[3], and Thomas Doll[4]

[1]Operational Analysis and Studies, Airbus Defence and Space, Immenstaad, GERMANY
[2]Simulations and Studies, Airbus Defence and Space, Immenstaad, GERMANY
[3]Army Concepts and Capabilities Development Centre, Bundeswehr, Cologne, GERMANY
[4]German Joint Support Command, Bundeswehr, Bonn, GERMANY

ABSTRACT

The growing complexity of modern warfare demands advanced AI-driven decision support for validating Operational Plans (OPLANs). This paper proposes a multi-agent reinforcement learning framework integrated into the ReLeGSim environment to rigorously test military strategies under dynamic conditions. The adoption of deep reinforcement learning enables agents to learn optimal behavior within operational plans, transforming them into "intelligent executors". By observing these agents, one can identify vulnerabilities within plans. Key innovations include: (1) a hybrid approach combining action masking for strict OPLAN adherence with interleaved behavior cloning to embed military doctrine; (2) a sequential training approach where agents first learn baseline tactics before evaluating predefined plans; and (3) data farming techniques using heatmaps and key performance indicators to visualize strategic weaknesses. Experiments show hard action masking outperforms reward shaping for constraint enforcement. This work advances scalable, robust AI-driven OPLAN validation through effective domain knowledge integration.

1 INTRODUCTION

The future of warfare is rapidly evolving, driven by digital, AI-powered command and control systems and the growing use of autonomous platforms. These advancements are accelerating the pace of operations, intensifying time pressures on military decision-makers. In response, modeling and simulation, augmented by advanced Artificial Intelligence (AI) techniques, are becoming essential components of next-generation decision support systems. Such systems hold the promise of enhancing situational understanding, threat evaluation and course-of-action analysis.

Recent advancements in AI research, exemplified by achievements such as DeepMind's AlphaGo (Silver et al. 2016) and AlphaStar (Vinyals et al. 2019) in the complex game of StarCraft II, demonstrate the potential of Deep Reinforcement Learning (DRL) to train agents capable of developing superior strategies. Unlike traditional Reinforcement Learning (RL), which often struggles with scalability and the complexity of large input spaces, DRL leverages deep neural networks to automatically learn and optimize complex representations (Mnih et al. 2015) from raw input data. This capability allows DRL to handle high-dimensional, continuous state spaces more effectively, making it particularly suitable for solving intricate problems such as those encountered in military strategy and operations. Importantly, DRL also enables a more flexible and adaptive approach to enforcing operational constraints compared to traditional, rule-based implementations. Whereas conventional simulations often hardcode behaviors – limiting flexibility and adaptability – our approach uses hard action masking to ensure agents adhere to OPLANs while preserving their ability to learn and discover novel, effective tactics.

This paper investigates the application of these techniques to a crucial aspect of military planning: the validation of OPLANs. Specifically, we explore how DRL agents can be trained to act as "intelligent executors" of the OPLAN against the opposing team which, instead, strictly follows the OPLAN. This setup

has been demonstrated to rigorously test OPLANs and identify potential weaknesses before they are deployed in real-world scenarios. The core of our work centers around a simulation environment, ReLeGSim (*Re*inforcement *Le*arning focused *G*eneric AI Training *Sim*ulation) (Doll et al. 2021), specifically designed for military engagements at the battalion level. Within ReLeGSim, a DRL agent can receive commands to control available units/companies or request fire support, while the simulation provides feedback – a "reward" – to evaluate and improve the agent's behavior through iterative training. This allows for a dynamic and evolving assessment of plan viability, beyond traditional tabletop exercises or wargaming.

Traditionally, validating OPLANs has relied on manual analysis, expert judgment and limited simulations. These methods are often time-consuming, resource-intensive and prone to human biases. Furthermore, they struggle to anticipate unforeseen consequences or emergent behaviors that may arise during execution. The use of AI agents offers a compelling solution to these limitations, providing a scalable and objective way of stress-testing OPLANs across a wide range of conditions. Our research builds on this premise, exploring not only the technical feasibility of training such agents but also the vital role of reward function design and the integration of domain-specific knowledge – represented by established military principles – into the learning process.

A key challenge investigated in this study is how to effectively incorporate established military doctrine, such as principles of maneuver warfare – protecting flanks, coordinating firepower and movement, establishing reserves and exploiting terrain – into the RL training process. Initial attempts at a strictly hierarchical RL approach, mirroring the command structure, proved difficult, as lower-level units struggled to balance obedience to higher-level commands with optimizing their own tactical situations. Instead, we adopted a sequential approach, first training an agent to achieve mission objectives without a pre-defined plan, then using the resulting behavior (identified through heatmaps revealing preferred attack paths and areas of conflict) to inform the creation of potentially effective OPLANs. This separation of planning and execution allows for a more nuanced evaluation, where the AI can objectively assess the inherent strengths and weaknesses of a given plan.

Furthermore, we address the critical issue of accelerating the RL training process. Recognizing that traditional methods can be computationally expensive, we explored techniques such as leveraging cloud computing resources and employing supervised learning approaches, specifically Interleaved Online Behavior Cloning (IOBC). IOBC utilizes rule-based agents – embodying established military tactics – to demonstrate desired behaviors to the RL agent, effectively jump-starting the learning process and improving performance, particularly in early training stages. This hybrid approach combines the adaptability of RL with the reliability of expert knowledge (Möbius et al. 2024). Finally, we highlight the importance of Verification and Validation (V&V) of such AI-driven systems, acknowledging the unique challenges posed by their stochastic nature and the need for rigorous testing and analysis.

This paper details the architecture of our AI-driven decision support framework, the experimental methodology employed and the key findings. It aims to demonstrate the potential of AI-agents, trained within a realistic simulation environment, to significantly enhance the robustness and effectiveness of operational planning by providing a less biased and comprehensive OPLAN validation capability.

2 RELATED WORK

2.1 AI and Simulation in the Military

This section reviews relevant research in military simulation, RL methods applicable to this domain and techniques for incorporating constraints into RL algorithms, specifically focusing on soft constraints via reward shaping and hard constraints via action masking.

Military simulation has a long history, initially focusing on wargaming and manual exercises to analyze tactics and logistics. Early simulations were largely analytical and relied on pre-defined models. Modern military simulations, however, are increasingly sophisticated, incorporating realistic environments, complex agent behaviors and detailed representations of weapons systems. These simulations serve various

purposes, including training, mission rehearsal and, crucially, operational planning and analysis, such as the military analysis simulation PAXSEM of the German Armed Forces (Kallfass and Schlaak, 2012).

The use of AI within military simulations has expanded significantly. Computer Generated Forces (CGF), AI-controlled entities populating the simulated environment, provide realistic opposition forces and allow for exploration of a wider range of scenarios (Taylor et al. 2009). However, traditional CGF often relies on rule-based systems that lack the adaptability and learning capabilities of modern AI agents. The work presented in this paper builds on this trajectory, aiming to leverage the power of reinforcement learning to create AI agents capable of not only reacting to dynamic situations, but also proactively evaluating and optimizing operational plans. The study described in van Oijen et al. (2023) highlights this shift, noting that the integration of AI is crucial for future decision support systems due to increasing time pressures and the need to assess a multitude of options.

2.2 Reinforcement Learning

Reinforcement learning provides a powerful framework for training AI agents to make sequential decisions in complex environments. The core principle of RL involves an agent learning through trial and error, maximizing a reward signal based on its actions (Sutton and Barto 2018). Several RL algorithms are particularly relevant to military applications.

Proximal Policy Optimization (PPO) and its asynchronous variant, Asynchronous Proximal Policy Optimization (APPO), are popular on-policy RL algorithms known for their stability and sample efficiency (Schulman et al. 2017). APPO, specifically used in the study described in Doll et al. 2021, leverages distributed computing to accelerate training, vital for complex military simulations.

Hierarchical Reinforcement Learning (HRL) aims to address the challenges of learning long-horizon tasks by decomposing them into sub-tasks (Li et al. 2019). This aligns well with the hierarchical nature of military command structures, where high-level objectives are broken down into lower-level actions. However, as highlighted in this paper, applying HRL directly to military scenarios has proven difficult, due to conflicts between the overall mission objective and optimizing localized sub-goals.

Wang et al. (2019) improved reinforcement learning methods by incorporating rule-based trajectory constraints, resulting in better performance of autonomous vehicles. Liu et al. (2023) transferred prior knowledge of an existing rule-based control policy by adding a behavior cloning term to regularize their online policy. Möbius et al. (2024) improved upon this and introduced the algorithm Curriculum Interleaved Online Behavior Cloning (IOBC).

Military operational planning often involves balancing multiple, potentially conflicting objectives – maximizing mission success, minimizing casualties, reducing collateral damage and conserving resources. Traditional RL often focuses on maximizing a single reward signal. Multi-Objective Reinforcement Learning (MORL) addresses this limitation by allowing agents to learn policies that optimize multiple objectives simultaneously (Roijers et al. 2018).

MORL algorithms typically aim to find a Pareto-optimal set of policies, representing the best possible trade-offs between the different objectives. Techniques include scalarization (combining objectives into a single reward using weights), vector-valued RL (learning separate policies for each objective) and preference-based RL (allowing a human operator to express preferences between different objectives) (Hejna and Sadigh 2023). Applying MORL to operational planning could enable AI agents to generate a diverse set of plans, each representing a different trade-off between competing priorities, allowing commanders to select the plan that best aligns with their strategic goals.

2.3 Constrained Reinforcement Learning

Imposing constraints is crucial in military applications, where certain actions or outcomes may be unacceptable regardless of their impact on the primary objective. There are two primary approaches to integrating constraints within reinforcement learning frameworks. The first is reward shaping, where penalties are incorporated into the reward function to discourage undesirable behaviors. This method

imposes a "soft" constraint, as the agent may still violate the constraints but will face penalties for doing so. While intuitive, reward shaping presents significant challenges. Crafting effective penalty functions requires meticulous design to prevent unintended consequences or suboptimal policies. A study by Möbius et al. (2023) demonstrated that over-reliance on reward shaping to enforce constraints can be ineffective, often leading the agent to prioritize reward maximization over adherence to operational plans.

The second approach, action masking, explicitly prevents the agent from taking actions that violate constraints (Achiam et al. 2017). This "hard" constraint method modifies the agent's action space, removing invalid actions for a given state. Hou et al. (2023) demonstrated success with this approach by enabling the AI agent to focus on optimizing within prescribed bounds. However, action masking can limit the agent's exploration and prevent the discovery of optimal strategies that may involve temporarily violating constraints, such as deviating from a predefined path in a military context to avoid threats or seize opportunities.

3 METHODS

3.1 The Simulation Environment "ReLeGSim"

ReLeGSim (shown in Figure 1) is a rasterized, Gymnasium-compatible simulation environment designed for developing and testing RL agents in complex, multi-agent scenarios. It supports single-agent, multi-agent and hierarchical RL, and is applicable to a wide range of civil and military use cases, including Intelligence, Surveillance and Reconnaissance (ISR) missions and joint battalion-level combat operations. The environment was presented at the NATO MSG-207 symposium (Möbius et al. 2023) and supports research in strategic AI for defense applications. It aligns with NATO's ongoing efforts, such as the SAS-181 task group, to explore reinforcement learning for decision-making in wargaming and operational planning.

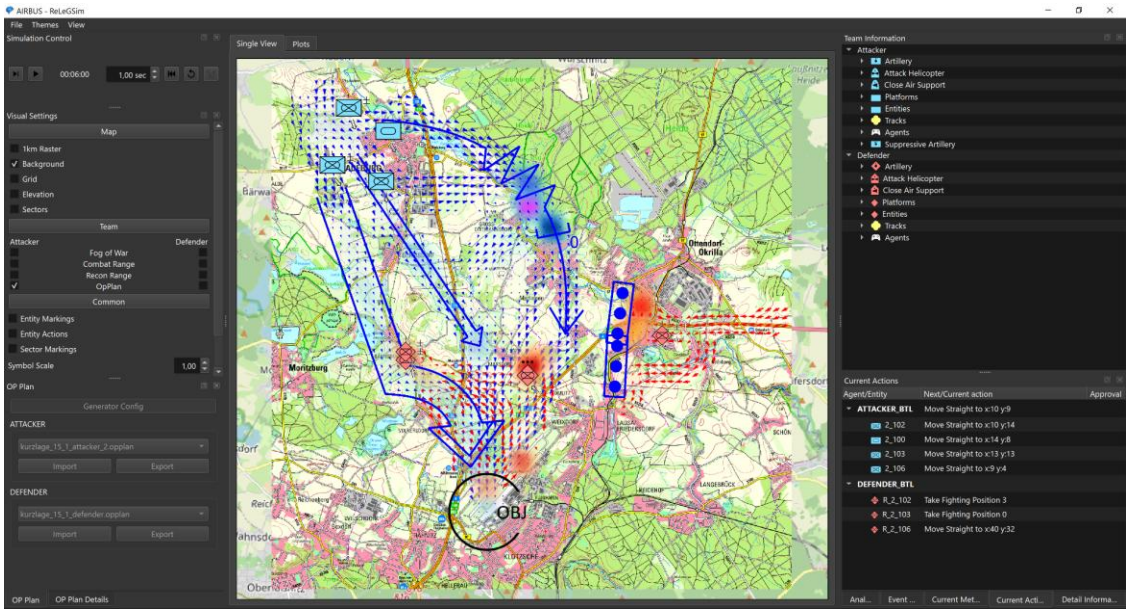

Figure 1: ReLeGSim expert-UI.

In ReLeGSim, agents learn to achieve objectives through sequences of actions such as movement, coordination and fire support requests. Users can define diverse actors, such as battalions, Unmanned Aircraft System (UAS) swarms, or individual units, with customizable properties and capabilities. The environment is extensible via Python plugins, allowing integration of domain-specific models (e.g., sensors or communication systems).

Agents operate in a dynamic, partially observable world, perceiving the environment through raw image and vector inputs. The large action space, long time horizons and delayed rewards present significant challenges for learning and planning. In adversarial scenarios, one agent acts as the attacker attempting to capture a target, while the defender must hold the position. Success requires strategic use of heterogeneous units, terrain awareness and anticipation of the opponent's actions.

Integrated within a broader Machine Learning Operations (MLOps) architecture, ReLeGSim includes a scenario generator that creates realistic 3D terrain from real-world data (elevation, satellite imagery, etc.), which is rasterized into terrain types for training. The framework supports AI vs. human gameplay, enabling benchmarking and evaluation. It also includes automated testing and customizable performance analysis tools for rigorous validation of trained agents.

3.2 ReLeGs-AI Architecture

The ReLeGs simulation employs a RL framework for its AI agents, built around the Gymnasium-Interface to facilitate compatibility with various RL algorithms. The architecture (shown in Figure 2) can be broadly divided into the simulation environment itself (ReLeGSim) and the RL-framework component. The simulation handles the core mechanics and environment interaction, while the RL framework manages agent learning and decision-making.

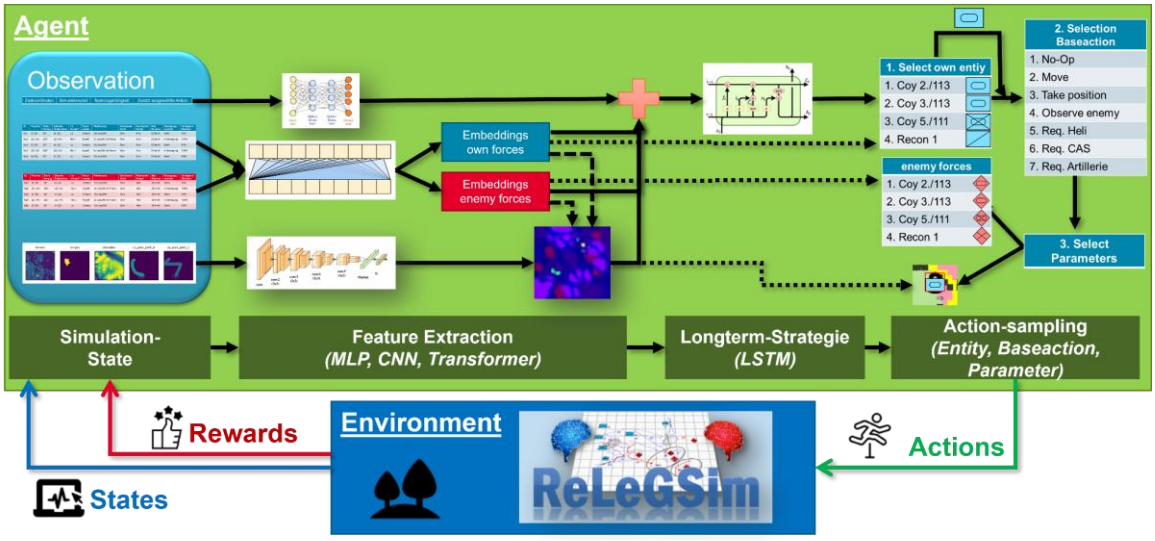

Figure 2: ReLeGs-AI architecture.

The core of the AI implementation utilizes Keras as a wrapper around PyTorch for building and training the neural networks. This allows for flexibility in potentially switching to other ML frameworks in the future. The training employs the APPO algorithm implemented within the Ray/RLlib distributed computing framework. Ray enables parallelization of the training process across multiple cores and GPUs, significantly accelerating learning.

The input to the AI model (left side of Figure 2) consists of three streams: a vector containing general simulation state information, tables containing data about units on both sides plus fire support units and a visual stream representing the terrain. Additional information, such as unit masks and unit positions, is provided to the model to perform masking and scattering operations, respectively. These streams are processed by different neural network components – a Multilayer Perceptron (MLP) for the vector data, transformer modules for units and fire support tables and a Convolutional Neural Network (CNN) followed by a residual network for the visual stream. The outputs are then aggregated and fed into a Long Short-Term Memory (LSTM) cell to capture temporal dependencies.

Action selection (right side of Figure 2) is a multi-stage process. First, the agent selects which of its own units to command. Then, it chooses a base action (move, request fire support, etc.). Finally, if required by the selected action, it specifies additional parameters (e.g., target coordinates for movement). A crucial aspect of this architecture is the implementation of action masking; only valid actions for a given unit in a given situation are available to the agent, simplifying learning and preventing invalid commands.

To facilitate OPLAN execution, the AI was configured to operate under strict control, adhering to a predefined plan instead of pursuing independent optimization. Initially, this was tested using soft constraints with a reward-based approach. However, after these attempts proved unsuccessful, we implemented strict action masking to enforce hard constraints. This hard masking blocks all actions that do not align with the OPLAN.

3.3 AI Decision Space

The action space in ReLeGSim is designed to be both flexible and learnable for the AI agent, allowing it to issue commands within the simulation environment. It's structured hierarchically, breaking down decision-making into three levels: unit selection, base action selection and parameter specification. First, the agent chooses which of the available friendly units, or companies, to control. This selection is informed by embeddings representing units' capabilities and current state. Once a unit is selected, the agent then picks a base action from a defined set including doing nothing, moving, assuming a defensive position, observing for enemies, or requesting fire support. Crucially, not all base actions are always available; the action space is dynamically masked to prevent the agent from attempting invalid commands based on the unit chosen and the current game situation – for example, it can't request fire support if no targets are in range. The base action selection is dependent on the selected unit as we add the embedding of this token to the state embedding.

The final stage involves specifying parameters for the chosen action. For a "Move" action, this means selecting target coordinates within a limited local area, constrained by terrain and any active operational plan. Going into a position requires choosing a suitable nearby position. "Observe" requires designating a target and "Request Fire Support" needs a visible enemy unit to be selected. The sampling of each parameter contains information about the selected unit and base action. This is obtained by adding the embedding of such tokens to the state embedding. A key feature of ReLeGSim is its ability to operate under pre-defined operational plans, or OPLANs. When an OPLAN is active, the action space is heavily constrained via hard masking, allowing only actions and parameters that align with the plan's directives (shown in Figure 3). This ensures that the AI follows the tactics imposed, enabling evaluation of the OPLAN's effectiveness, rather than the agent's independent decision-making.

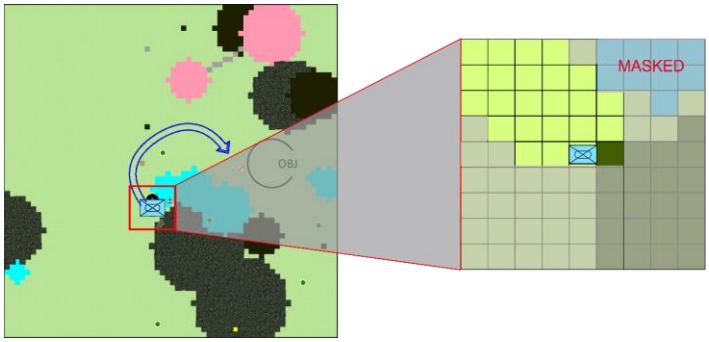

Figure 3: Masking of movements.

Early development explored a Natural Language Processing (NLP) based action space, hoping for increased flexibility (Möbius et al. 2022). However, this approach proved problematic. The vastness of the resulting action space hindered learning and the high rate of invalid action sequences demanded complex

fallback systems. Implementing dynamic masking within an NLP framework also proved difficult, ultimately leading to its rejection in favor of the current hierarchical, masked action space.

3.4 Reward Design

The development of an effective reward function is one of the most critical aspects in training reinforcement learning agents, particularly in the context of military decision support. The reward function must align with operational goals and provide clear, actionable feedback to the agent during training. In the context of the ReLeGSim, the reward function plays a pivotal role in guiding the agent to make strategic decisions that reflect military tactics and objectives.

The primary challenge in designing the reward function lies in balancing multiple, often conflicting, objectives. For instance, the agent must be incentivized to achieve its mission goals, such as capturing a target area, while simultaneously minimizing its own losses and adhering to operational plans. This requires a reward function that is both comprehensive and finely tuned to the specific requirements of the simulation.

Initially, the reward function was designed to be simple and straightforward, focusing on key metrics such as proximity to the target, successful capture of the target area and the safeguarding of friendly units. The agent received a small positive reward for each time step it moved closer to the target and a small negative reward for moving away from it. Upon successfully capturing the target area within the allotted time, the agent received a large positive reward, which was further adjusted based on the number of surviving friendly units. This basic structure ensured that the agent had clear incentives to advance towards and secure the target while maintaining its forces.

However, this initial design proved insufficient for more complex scenarios where the agent needed to adhere to specific operational plans. To address this, additional rewards were introduced to encourage the agent to follow the designated paths and tactical maneuvers outlined in the OPLAN. For example, the agent received a positive reward for staying within a certain distance of the assigned path and a negative reward for deviating significantly from it. This helped ensure that the agent's actions aligned with the strategic intent of the OPLAN, even if it meant sacrificing some immediate tactical advantages.

Despite these enhancements, the complexity of the reward function posed significant challenges. A reward function with over 200 lines of complex code became difficult to maintain and understand, leading to issues in transparency and predictability. Moreover, optimizing multiple, sometimes contradictory, reward criteria required careful tuning and balancing. For instance, the agent might be incentivized to minimize its own losses, but this could conflict with the need to aggressively pursue the enemy and capture the target area.

To mitigate these challenges, the concept of hard masking was introduced. Instead of relying solely on the reward function to guide the agent, hard masks were applied to restrict the agent's action space to only those actions that were consistent with the OPLAN. This approach significantly reduced the complexity of the reward function and provided a clearer, more direct way to enforce adherence to the operational plan. For example, when the agent was required to follow a specific path, any actions that would cause it to deviate from that path were masked out, ensuring that the agent could not choose them.

The use of hard masking also allowed for a simpler and more intuitive reward function. The agent no longer needed to balance multiple, potentially conflicting, reward signals. Instead, it focused on achieving the primary mission objectives, such as capturing the target area, while naturally adhering to the constraints imposed by the OPLAN. This led to more predictable and controllable behavior, making it easier to evaluate the effectiveness of different OPLANs and to refine the agent's performance.

In summary, the design of the reward function in ReLeGSim evolved from a simple, goal-oriented structure to a more sophisticated system that incorporated operational constraints through hard masking. This approach not only simplified the reward function but also enhanced the agent's ability to execute complex military operations effectively. By combining clear, actionable rewards with strict action constraints, the reward function played a crucial role in training the RL agent to make strategic decisions that aligned with military objectives and operational plans.

3.5 Operational Plans

ReLeGSim supports the creation and execution of operational plans at the battalion level, allowing commanders to define and optimize OPLANs that are then executed by AI agents. The process for the decision support for the commander is described in Figure 4. The system's graphical user interface is a key component, enabling users to intuitively draw attack arrows, place units and define objectives on a map. Users can specify parameters for each OPLAN, including flanking attacks (right or left), types of attacks (main, supporting, or suppressing), the use of Joint Fire Support (JFS) and the utilization of specific positions to reinforce unit strength. Once an OPLAN is defined, AI agents follow the specified parameters, with certain actions masked to prevent deviation from the plan. For instance, units designated to follow a specific path are not allowed to choose actions leading them off course (shown in Figure 3) and AI agents are restricted to local movement coordinates to ensure adherence to the OPLAN.

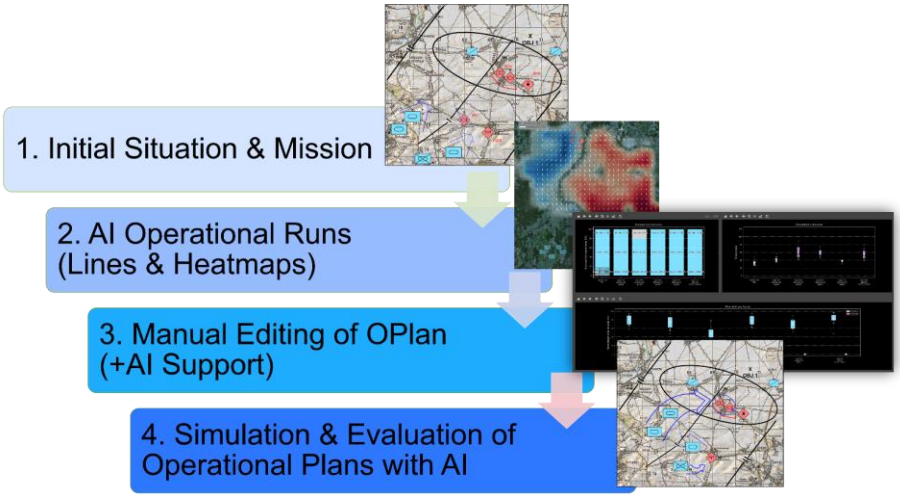

Figure 4: Creating operational plans with AI support.

Data farming, a design of experiments driven analysis methodology (Horne and Meyer 2024), is employed to evaluate OPLAN effectiveness through extensive simulation runs, generating large datasets with varying initial conditions. Each simulation run measures Key Performance Indicators (KPIs) such as mission success rate, number of losses and operation time, providing insights into the performance of the OPLAN under different circumstances.

Heatmaps visualize the results of these simulations, showing the most frequently used paths, strategic routes and areas of high activity. They highlight key battle zones, the effectiveness of different tactics and the impact of weapon systems and resulting casualties. Heatmaps facilitate comparative analysis between different OPLANs, helping users identify weak points and refine their strategies to improve mission success rates and reduce losses. Users can interactively create OPLANs by drawing attack paths, placing units and defining objectives and customize plans with detailed parameters. Real-time monitoring through an estimation of the KPIs via a surrogate model allows users to track the progress of the OPLAN and make informed adjustments. The surrogate model is designed to estimate the resulting outcome of a provided OPLAN in a given scenario, The final Plan validation is done through extensive data farming analysis and the use of heatmaps and KPIs provide valuable insights, enabling users to understand the impact of their OPLANs and ensure they are robust and effective in real-world scenarios.

4 RESULTS

The integration of AI agents into the validation of operational plans has demonstrated significant potential in enhancing the robustness and effectiveness of military decision-making. Through the use of a multi-agent reinforcement learning framework within the ReLeGSim simulation environment, we have

successfully developed a system capable of rigorously testing OPLANs and identifying potential vulnerabilities. The results of our experiments highlight the feasibility and benefits of this approach, while also shedding light on the challenges and areas for future improvement.

In our experiments, we utilized the ReLeGSim environment, equipped with a deep reinforcement learning agent trained using the Asynchronous Proximal Policy Optimization (APPO) algorithm. The agent was trained through a curriculum approach with levels ranging from 0 to 13, each characterized by specific configurations that progressively increased in complexity (as shown in Figure 5). For instance, the initial level featured a single attacker against no defenders, with subsequent levels introducing more attacker and defender units. Notably, when the attacker team consisted of more than three units, one unit could be designated as a reserve without any operational plan. Each other unit was assigned an OPLAN.

This approach enabled the agent to learn and execute sophisticated strategies, similar to those observed in advanced AI systems like AlphaStar. The evaluation metrics reported in Figure 6 demonstrated the agent's ability to handle diverse and dynamic situations that closely mimic real-world military engagements.

One of the key findings of our study is the effectiveness of the sequential approach to training the AI agent. Initially, the agent was trained to achieve mission objectives without a predefined operational plan,

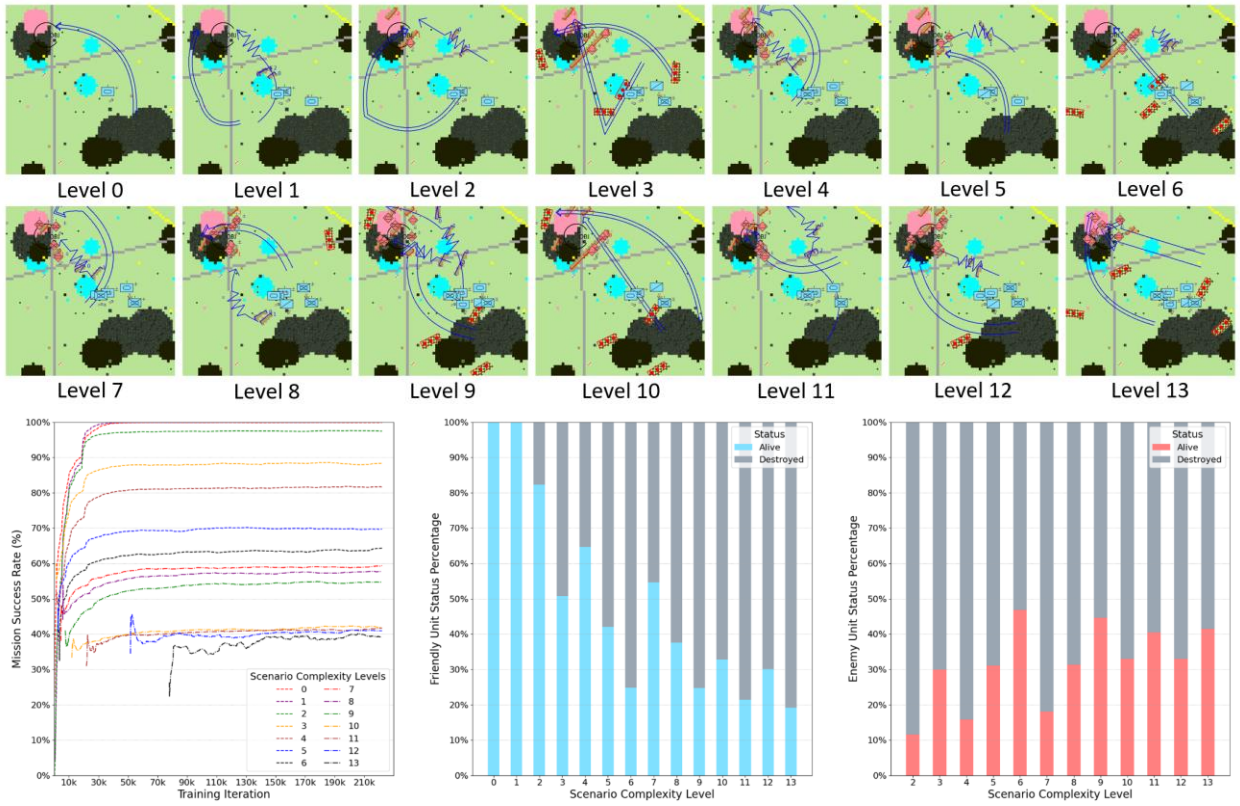

Figure 6: Warfighting metrics. Mission success rate during training across complexity levels (left). Status of friendly units per complexity level (center). Status of enemy units per complexity level (right). It took 12 days to reach the final level (78k iterations) using an NVIDIA RTX 4090 GPU and 60 CPUs.

allowing it to develop a baseline understanding of effective tactics and strategies. This initial training phase revealed preferred attack paths and areas of conflict through heatmaps, which were then used to inform the creation of potentially effective OPLANs. By separating the planning and execution phases, we were able to create a more nuanced evaluation process, where the AI could objectively assess the strengths and

weaknesses of a given plan. This approach not only enhanced the agent's adaptability but also provided valuable insights into the operational dynamics of different scenarios.

Another significant finding is the importance of action masking in enforcing operational constraints. Early attempts to use soft constraints, such as reward shaping, to guide the agent's behavior were found to be ineffective and often led to suboptimal policies. By implementing hard action masking, we were able to ensure that the agent adhered strictly to the operational plan, focusing its efforts on optimizing within the defined boundaries. This approach significantly improved the reliability and accuracy of the simulation results, allowing for a more focused evaluation of the OPLAN's effectiveness.

The use of heatmaps and key performance indicators further enhanced the validation process. Heatmaps (see Figure 7) provided a clear and intuitive visualization of the AI's behavior and the outcomes of the OPLAN, highlighting strategic routes, areas of high activity and key battle zones. KPIs such as mission success rate, number of losses and operation time were measured for each simulation run, offering quantitative insights into the performance of the OPLAN. These tools facilitated comparative operational analysis between different OPLANs, enabling users to identify weak points and refine their strategies iteratively. The combination of heatmaps and KPIs provided a comprehensive and actionable assessment of the OPLAN's effectiveness, significantly improving the quality of operational planning.

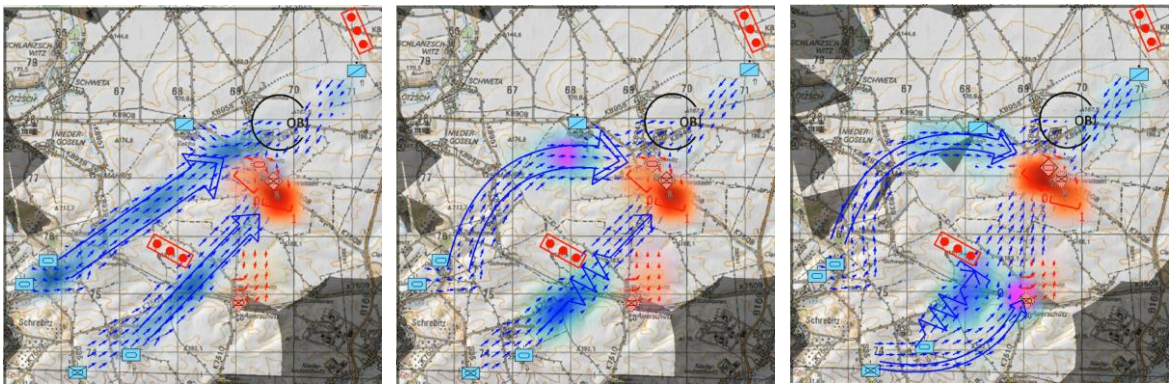

Figure 7: Comparison of three different OPLANS.

5 CONCLUSION AND WAY FORWARD

In conclusion, the results of our study provide strong evidence for the potential of AI agents in validating operational plans. The ReLeGSim environment, coupled with advanced reinforcement learning techniques, offers a powerful tool for enhancing the robustness and effectiveness of military decision-making. By addressing the identified challenges and continuing to refine the system, we can move closer to realizing the full potential of AI in operational planning and decision support. The way forward involves further research into efficient training methods, the integration of domain-specific knowledge and the development of scalable and adaptable systems capable of handling a wide range of operational scenarios.

Our study also uncovered several challenges that need to be addressed in future research. One of the primary challenges is the computational cost associated with training deep reinforcement learning agents. While the use of distributed computing frameworks like Ray/RLlib has significantly accelerated the training process, the computational requirements remain substantial, especially for complex, high-resolution simulations. Future work should explore more efficient training methods to further reduce training times.

Another challenge is the integration of domain-specific knowledge into the RL training process. While the sequential training approach and action masking have improved the agent's adherence to operational plans, there is still room for enhancing the agent's understanding of established military doctrines and principles. Future research should focus on developing more sophisticated reward functions and training methodologies that incorporate a broader range of military expertise and tactical knowledge. This could involve the use of hybrid approaches, combining reinforcement learning with Large Language Model

(LLM)-based systems with doctrine and expert knowledge, to create more robust and adaptable AI agents. We are exploring the possibilities of connecting a LLM to a military simulation in the "KITCH" (AI-Tactic-Chat) study with the German armed forces.

Additionally, the scalability and generalizability of the system to different operational contexts and scenarios remain areas for further investigation. While our experiments have demonstrated the effectiveness of the system in specific scenarios, the ability to generalize to new and unseen environments is crucial for practical deployment. Future work should explore the transferability of learned policies and the development of adaptive learning mechanisms that can quickly adjust to changing conditions. This will be essential for the emerging topic of SDD (Software Defined Defence), which is an important topic for the German armed forces. As a concrete application, the proposed system will be extended to support mission planning for autonomous UAV swarms within the KITU (AI for Tactical UAVs) project, further demonstrating its potential in next-generation defense technologies.

ACKNOWLEDGMENTS

In memoriam of LTC Dr. Dietmar Kunde from the German Army Headquarters, whose enduring mentorship, guidance and dedication to the practical application of AI continue to inspire and drive our research. In addition, we would like to thank Matthias Flock, Ruben Jacob, Riccardo Paolini and Jonas Wild for their valuable contributions to this work. We also extend our gratitude to Prof. Oliver Rose from the University of the Bundeswehr Munich for his invaluable contribution in advancing our research activities.

REFERENCES

Achiam, J., D. Held, A. Tamar, and P. Abbeel. 2017. "Constrained Policy Optimization". *arXiv preprint arXiv:1705.10528.*

Doll, T., J.-W. Brendecke, M. Behm, and D. Kallfass. 2021. "From the Game Map to the Battlefield – Using DeepMind's Advanced AlphaStar Techniques to Support Military Decision-Makers". In *NATO MSG-184: Towards Training and Decision Support for Complex Multi-Domain Operations*, October 21st-22nd, Amsterdam, Netherlands.

Hejna, J., and D. Sadigh. 2023. "Inverse Preference Learning: Preference-based RL without a reward function". In *Proceedings of the 37th International Conference on Neural Information Processing Systems*. https://dl.acm.org/doi/10.5555/3666122.3666947.

Hou, Y., X. Liang, J. Zhang, Q. Yang, A. Yang, and N. Wang. 2023. "Exploring the Use of Invalid Action Masking in Reinforcement Learning: A Comparative Study of On-Policy and Off-Policy Algorithms in Real-Time Strategy Games". *Applied Sciences* 13(14): 8283 https://doi.org/10.3390/app13148283.

Horne, G., and T. Meyer. 2004. "Data Farming: Discovering Surprise". In *2004 Winter Simulation Conference (WSC)*, 171–180 https://dl.acm.org/doi/10.5555/2429759.2430057.

Kallfass, D., and T. Schlaak. 2012. "NATO MSG-088 Case Study Results to Demonstrate the Benefit of using Data Farming for Military Decision Support". In *2012 Winter Simulation Conference (WSC)*, 2481–2492 https://dl.acm.org/doi/10.5555/2429759.2430057.

Li, S., R. Wang., M. Tang, M., and C. Zhang, 2019. "Hierarchical Reinforcement Learning with Advantage-Based Auxiliary Rewards". *Advances in Neural Information Processing Systems 32.*

Liu, H.Y., B. Balaji, R. Gupta, and D. Hong. 2023. "Rule-Based Policy Regularization for Reinforcement Learning-based Building Control". In *Proceedings of the 14th ACM Conference on Future Energy Systems*, June 21st-23rd, Orlando, USA, 242–265.

Mnih, V., K. Kavukcuoglu, D. Silver, A. Rusu, J. Vaness, M. Bellemare, et al. 2015. "Human-level Control Through Deep Reinforcement Learning". *Nature* 518: 529–533.

Möbius, M., D. Kallfass, T. Doll, and D. Kunde. 2022. "AI-based Military Decision Support Using Natural Language". In *2022 Winter Simulation Conference (WSC),* 2082–2093 http://dx.doi.org/10.1109/WSC57314.2022.10015234.

Möbius, M., D. Kallfass, T. Doll, and D. Kunde. 2023. "Incorporation of Military Doctrines and Objectives into an AI Agent Via Natural Language and Reward in Reinforcement Learning". In *2023 Winter Simulation Conference (WSC),* 2357–2367 http://dx.doi.org/10.1109/WSC60868.2023.10408462.

Möbius M., K. Fischer, D. Kallfass, S. Göricke and T. Doll. 2024. "Curriculum Interleaved Online Behavior Cloning for Complex Reinforcement Learning Applications,". In *2024 Winter Simulation Conference (WSC)*, 1990–2001 https://dl.acm.org/doi/10.5555/3712729.3712895.

Roijers, D., L. Zintgraf, P. Libin, and A. Nowe. 2018. "Interactive Multi-objective Reinforcement Learning in Multi-armed Bandits for Any Utility Function". In *Proceedings of the Adaptive and Learning Agents Workshop (ALA-18) at AAMAS.*

Schulman, J., F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. 2017. "Proximal Policy Optimization Algorithms". *arXiv preprint arXiv: 1707.06347.*

Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, et al. 2016. "Mastering the Game of Go with Deep Neural Networks and Tree Search". *Nature* 529(7587): 484–489.

Sutton, R. S., and A. G. Barto. 2018. "Reinforcement learning: An introduction". *MIT Press*.

Taylor, A., Abdellaoui, N., and Parkinson, G., 2009. "Artificial Intelligence in Computer Generated Forces: Comparative Analysis". In *The Huntsville Simulation Conference (HSC 2009)*, 199–205. Huntsville, AL, USA, October 27–29.

Vinyals, O., I. Babuschkin, W.M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung. et al. 2019. "Grandmaster Level in StarCraft II using Multi-Agent Reinforcement Learning". *Nature* 575(7782): 350.

Van Oijen, J., P. de Marez Oyens, 2023. "Empowering Military Decision Support through the Synergy of AI and Simulation". In *NATO MSG-207: Simulation: Going Beyond the Limitations of the Real World*, October 19th–20th, Monterey, CA, USA.

Wang, J., Q. Zhang, D. Zhao, and Y. Chen. 2019. "Lane Change Decision-Making through Deep Reinforcement Learning with Rule-Based Constraints". In *International Joint Conference on Neural Networks (IJCNN),* July 14th -19th, Budapest, Hungary.

AUTHOR BIOGRAPHIES

MICHAEL MÖBIUS is expert in Operational Analysis and AI-Enabled simulation at Airbus Defence and Space in Germany, leading AI and software projects in the "Operational Analysis and Studies" department. His research focuses on large language models (LLMs), stochastic simulation, autonomous systems, UAVs, reinforcement learning, software defined defence and operational analysis. His email address is michael.möbius@airbus.com.

DANIEL KALLFASS is senior expert in 3D simulation of System of Systems at Airbus Defence and Space. With over 20 years of experience in defense and security research, he specializes in simulation-based operational analysis and decision support using AI and in particular deep reinforcement learning and GenAI. His email address is daniel.kallfass@airbus.com.

LTC STEFAN GÖRICKE, of the German Army Concepts and Capabilities Development Center, focuses on constructive simulation and AI. He holds dual Master's degrees in Economics from Helmut Schmidt University and in Modeling and Simulation from the Naval Postgraduate School in Monterey, California, USA. His email address is stefangoericke@bundeswehr.org.

LTC THOMAS DOLL is a German Army officer responsible for conducting military analyses and studies at the German Joint Support Command. His main areas of interest are modeling and simulation, unmanned systems, and artificial intelligence. He studied electrical engineering at the University of the German Armed Forces and earned his Master of Science from the Naval Postgraduate School in Monterey, California, USA, in 2004. His email address is thomasmanfreddoll@bundeswehr.org.