Evaluation of AI Agent Accelerated Cyber Operations Planning

Pietari Sarjakivi and Panu Moilanen
University of Jyväskylä, Finland

pietari@sarjakivi.fi
panu.moilanen@jyu.fi

Abstract: As societies become increasingly digital, they are more exposed to cyber threats that have the potential to harm human life and damage critical infrastructure and other assets. To counter these fast-paced threats, Defensive Cyber Operations (DCO) leaders must enhance their capabilities for rapid decision-making and response. Artificial Intelligence (AI), as a radical levelling technology, has the potential to accelerate DCOs; however, the existing solutions frequently focus on narrow technical use-cases and lack emphasis on the leadership dimension of DCO. The purpose of this paper is to address that gap by researching how AI can accelerate one of the most relevant DCO use-cases identified in the author's earlier research, course of action recommendation, especially in operations planning. The study is based on case study methodology, where AI agent-generated operations plans are compared to real-life DCO operations plans made by leading experts in the world's most complex defensive cyber exercise, NATO Locked Shields 2025. The study focuses on two of the most critical decisions a DCO leader needs to make during the primary process of DCO: Prioritization of defended capabilities and assets, and the right resourcing and allocations. The selected exercise provided an excellent platform for this study to compare multiple human-made plans to machine-made plans, as 17 world-class blue teams were given the same exercise scenario and operation order. As a result, this paper demonstrates that with proper architecture and context engineering, AI can significantly accelerate DCO leaders' decision-making in operations planning, while human-machine teaming is still needed to navigate a complex operating environment where cyber operations are typically conducted. The main contributions of this paper are 1. evaluation of an AI agent's performance in DCO operations planning in comparison to human experts, and 2. construction of a reference architecture for the DCO planning agent. Future research can be built to improve the results of the reference architecture. As AI's capabilities are developing rapidly, it is expected that the capabilities of autonomic AI agents will increase.

Keywords: Defensive cyber operations, Artificial intelligence, Operations planning, AI agent, Context engineering

1. Introduction

With the world digitalizing, societies are becoming more dependent on their digital infrastructure. Simultaneously, cybercrime increases year-by-year and targets societies essential services, like critical manufacturing and healthcare (Federal Bureau of Investigation, 2025). Cyber Operations (CO) are becoming an increasingly integral part of hybrid warfare, with governments' ability to deny their participation on cyberattacks. Critical infrastructure providers, who are often private companies (Azrilyant et al., 2022), have difficulties defending their assets against advanced attackers, and therefore, governments need to provide assistance in the form of Defensive Cyber Operations (DCO) to recover from a cyber incident, like in Minnesota 2025 (City of Saint Paul, 2025). To enable effective DCOs performed by both military and civilians, a joint operations model is essential.

Artificial Intelligence (AI) is a radical leveling technology: it makes cyberwarfare fighting capabilities accessible to every nation and terrorist organization. Therefore, it has the potential to disrupt linear development in multiple domains, including cybersecurity (Snow, 2015). AI can improve COs in at least 22 use cases (Sarjakivi, 2025), and the rapid development of Large Language Models (LLMs) has enhanced AI agent maturity to a level where agents have the capability to autonomously exploit zero-day vulnerabilities in real-life systems with just a few-dollar cost (Fang et al., 2024). AI agents extend LLMs' reasoning and interpretation skills by synergizing Reasoning and Acting (ReAct) skills (Yao et al., 2023). Context engineering, a systematic approach exceeding the limitations of prompt engineering, is used for improving LLM performance (Pajo, 2025).

This study focuses on evaluating AI agents' capabilities to accelerate DCOs. The paper is organized as follows: Section 2 introduces the research methodology, section 3 presents the results for research questions, and section 4 concludes the study with future research topics.

2. Methodology

2.1 Research Methodology

This study utilizes a case study approach and consists of comparative content analysis and structured expert interview phases. A case study is an especially suitable methodology for this research as it focuses on "how" questions in contemporary phenomena within a real-life context where the researcher has little control over the events (Yin, 2003). *Course of Action (COA) recommendation* is the third most important AI use cases

accelerating COs, and, as the other high ranking use cases, namely *information sharing and reporting*, and *data fusion, enrichment, and visualization*, are relatively generic, this CO focused use case was selected for the evaluation. According to the same study, the most important decisions a DCO leader needs to make within the primary process are *prioritization of defended capabilities and assets*, and *the right resourcing and allocations* (Sarjakivi, 2025).

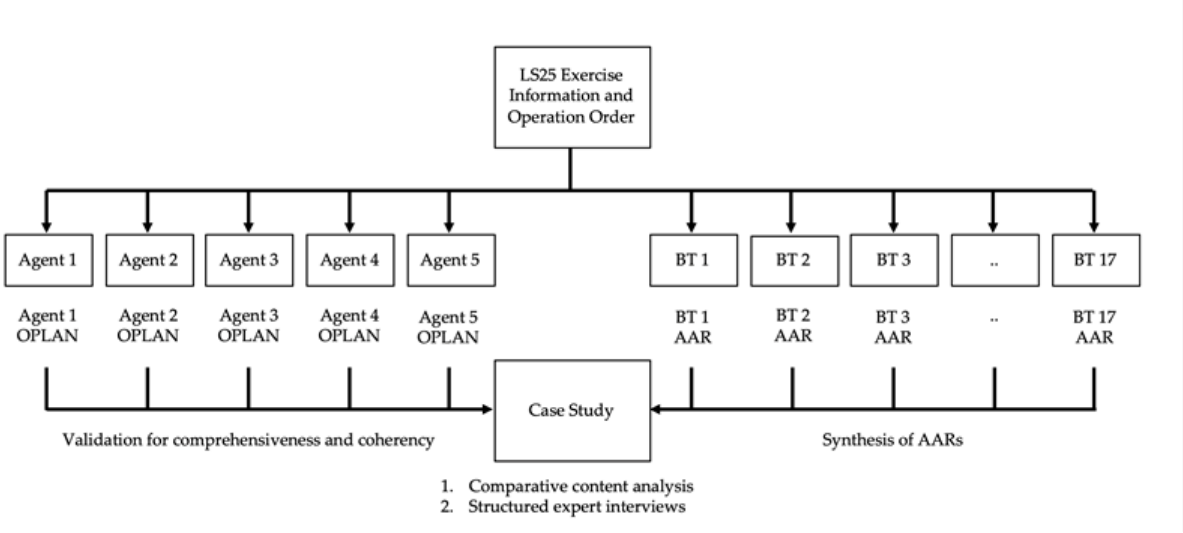

Figure 1: Research methodology utilized in this case study

As shown in Figure 1, the study was conducted by comparing AI agent-generated operations plans to real-life DCO operations plans made by human experts in NATO Locked Shields 2025. Locked Shields is the world's most complex cyber defense exercise, where over 4000 people from 41 nations defend a fictional country's military systems and critical infrastructure from cyber-attacks (NATO Cooperative Cyber Defence Centre of Excellence, 2025). In the exercise, each of the 17 world-class Blue Teams (BT) were given the same exercise information with scenario, rules of engagement, and the Operations Order (OPORD) to defend the environment. Although the BT's Operations Plans (OPLAN) were not available for comparison, most of the teams explain their approach in the After Action Report (AAR), from which the most critical decision can be extracted. Together with its scale and accessibility of AAR information, this exercise provided an excellent platform for comparison of human- and machine-made OPLANs.

To understand the impact of architecture and context engineering, 5 increasingly capable agents were created.

- Agent 1: Simple prompting.
- Agent 2: Advanced prompting using best practices
- Agent 3: Advanced prompting and Retrieval-Augmented Generation (RAG) datastore.
- Agent 4: Advanced prompting, RAG datastore, and manually injected context into the system prompt.
- Agent 5: Advanced prompting, RAG datastore, manually injected context into system prompt, and response improved with the recommendation of a separate Wargamer tool.

Each agent was tested with 3 LLM models, each possessing unique advantages and limitations, such as the needed computing power and the possibility of operating in offline environments. OpenAI's latest and highest performing model, GPT-5, was selected based on its reference test results (OpenAI, 2025b) to represent high-end, centrally hosted closed source models. Open AI's GPT-OSS-120b, which performs in similar level of closed source models like o4-mini (OpenAI, 2025a), was selected as heavy weight open source model that can be run offline as long as dedicated hardware is available, and DeepSeek's latest open source reasoning model, DeepSeek-R1-0528, as a high performing alternative to OpenAI's models (DeepSeek, 2025). As AI agents output different responses with each execution, each agent-model combination was executed 3 times, and the most comprehensive and coherent response was passed to further analysis. In total, 15 responses were analyzed. In the test setup, all the data and processing were done in a purpose-built, high-security environment to ensure secure data handling requirements.

In the first phase, comparative content analysis was performed by manually extracting key elements from AI agent responses and comparing them against the synthesized BT AARs results. The second phase was structured expert interviews with 3 experienced professionals who have experience of leading their nations'

Locked Shields BTs and creating the OPLAN for their teams. Experts evaluated agent responses by scoring the AI-generated OPLAN on a scale of 1-10 and answering the following questions with a scale of Yes (1) – Partially (0,5) – No (0).

- Is this plan executable?
- Is the prioritization of defended capabilities and assets reasonable?
- Are the resourcing and allocations reasonable?

2.2 Research Questions

This study aims to answer the following questions:

- RQ1: Can AI agents improve DCO operations planning?
- RQ2: What kind of context engineering and architecture are required for a DCO AI agent?

3. Results of Research Questions

3.1 RQ1: Can AI Agents Improve DCO Operations Planning?

As shown in Figure 2 below, the overall results of expert analysis indicate that AI agents improve DCO operations planning. Agent 4, which performed the highest, OPLAN scored 7/10 and got 80% from the executability measurement. Although the results are not at a sufficient level for autonomous operations, an AI agent can significantly accelerate operations planning. According to the results, the performance of different models and agents varies heavily, while GPT-5 and gpt-oss-120b seem to perform the highest. Also, the inconsistency between executions remained high throughout the evaluation, highlighting the need of Human-Machine Teaming (HMT), where human operators' analysis of the agents' responses and steer agent response generation to match the needs of a complex operating environment where COs are typically conducted.

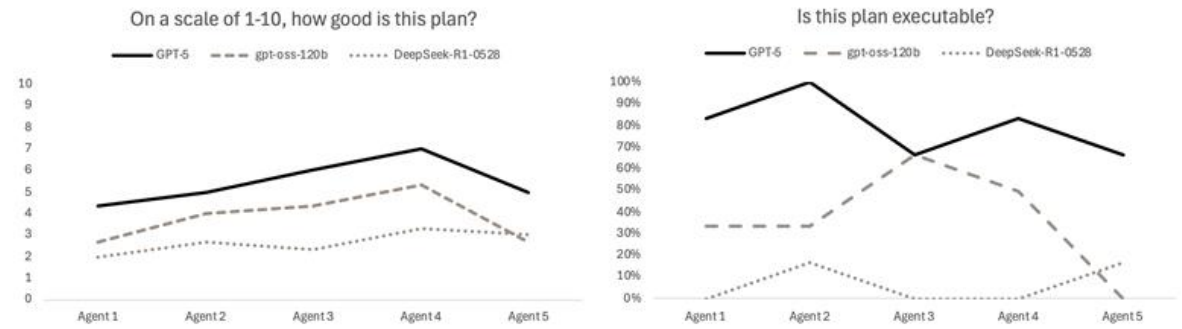

Figure 2: Expert analysis results on the overall quality of AI-generated OPLAN

On specific decisions of *prioritization of defended capabilities and assets*, AI agents seem to perform relatively good, as shown in Figure 3. Agent 4 with the GPT-5 model got a 100% rating from experts, while others performed relatively well. It is noticeable that all other agents hallucinated some of the defended capabilities and or provided incomplete lists. When agent 4's response was given to agent 5 to wargame and improve, the agent 5 focused on partially hallucinated technical details and lost the complete listing of defended capabilities. This is assumed to be caused by overwhelming context and loss of focus on the original task.

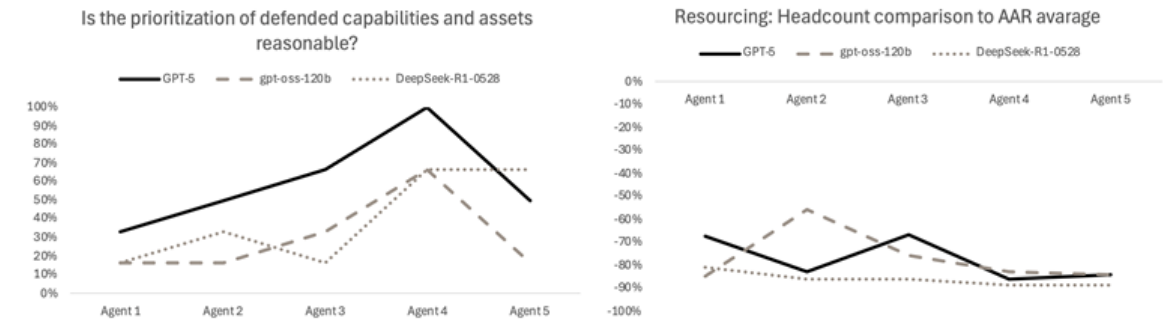

Figure 3: Agents' performance on the most important decisions

The act of choosing *the right resourcing and allocations* was proven to be a difficult task for the agents. As shown in Figure 3, agents choose team size on average 85% below the average of Locked Shields 2025. This variation can be partially explained by the fact that LLMs' training data is based on historical data, while the number of team members in Locked Shiels has over ten folded during the last ten years (NATO Cooperative Cyber Defence Centre of Excellence, 2025; Smeets, 2022). Also, the human-made plan might be impacted by external factors, like the economy and the availability of specialists. According to the AAR synthesis, a larger number of specialists does not guarantee success in the exercise, and in fact, high-performing BTs' participant count was 23% below the average.

Only 33% of AI agent-created organizations were utilizing matrix or defended capability-based team structures, which was the most utilized approach in Locked Shields 2025. Over 93% of agent-generated organizations were built to be low hierarchy, which is contradicting to real-life BT structures, where over 81% utilized a multi-layer organization structure with operations leaders between blue team commanders and squad leaders leading specialists. One contributing element to this variation can be a substantial difference in headcount, as bigger organizations usually need a higher level of hierarchy.

To conclude the study in a reasonable timeframe and make the results comparable, certain limitations had to be made. Agents' technical implementation and prompt tuning were the same for all the models and, therefore, not optimized for any of them. BT's Standard Operating Procedures (SOPs) were not given to the agent, as they vary between BT. Data captured from AARs didn't provide a complete understanding of BTs' OPLANs, and even if they had provided, a single right answer does not exist. While the number of experts analysing AI agents' responses was limited, their analysis was relatively cohesive. Even with limitations identified above, the authors believe the evaluation gives a trustworthy answer to AI agents' DCO operations planning capabilities.

3.2 RQ2: What Kind of Context Engineering and Architecture are Required for a DCO AI Agent?

This construction is built by applying AI agent best practices on Sarjakivi et al's (2025) DCO decision-making model. Reference implementations like military COA creation (Goecks & Waytowich, 2024), disaster response (Goecks & Waytowich, 2023), and information operations (Kereopa-Yorke, 2024) have been reviewed, and lessons learned from those implementations are adopted into this implementation.

3.2.1 Human-machine teaming and right expectations

Although in the academic literature, AI's role in accelerating COs is often seen as an independent agent (39%) (Sarjakivi, 2025), HMT, with its objective to find optimal performance of the team, builds on collaboration and interdependent teamwork of human and machine operators (e.g., AI agents) (Johnson & Bradshaw, 2021). Festor et al (2021) defined 5 levels of autonomy in safety-critical decision-making, where machine operators' role incrementally increases from advisor to independent decision maker while human operators' role decreases from decision maker to assurance and backup role, and eventually drops out of the decision-making loop completely. According to the situational leadership model, human operators' maturity in making decisions without assistance from more experienced practitioners is not universal but dependent on the situation and decision at hand (Hersey et al., 2013). The same model should be applied to machine operators' decision-making. Similarly to delegating decision-making to a junior colleague, the key is to understand how much errors and delays the particular decision tolerates, what the cost of a wrong decision is, and can we implement supporting controls or mitigations in case of a wrong decision made. Additionally, the situation where a decision is made impacts on the autonomy that should be granted to machine operators, and therefore, autonomy should be seen more as a dynamic slider controlled by an experienced human operator.

In the early levels of autonomy, machine operators pre-process and analyse the data needed for decision-making, increasing the speed of the decision-making loop. The scale and speed provided by the machine operator enable more thorough analysis with an extended number of alternative COAs analysed within the same time window provided by the frame operation. With mature enough machine operators, the current dependency on experienced human operators decreases, providing organizations the possibility to conduct multiple overlapping operations simultaneously with fewer or less experienced human operators. According to the results of RQ1 in section 3.1, human cognition is needed to verify the results of machine created plans. Human operators should focus on problems caused by physical and human elements in the plan execution, like extensive delays in responses, inaccuracy in communications, and lack of visibility the events in the real life.

3.2.2 Context engineering and safeguarding best practices

Improving the results of AI agents' performance requires several considerations.

Model. Each model has their unique performance characteristics, and it is critical to choose the right model that fits the purpose (Mbaiossoum et al., 2025). Multimodal models support different data types such as audio and images, base models are suitable for tasks requiring generic knowledge, and fine-tuned models excel on specific tasks. Chat models are optimal for conversation and writing tasks, while reasoning models are better at problem-solving and decision-making. An agent can be built to utilize different models based on the task at hand, for example, using a lightweight model for simple and pre-processing related tasks and premium model for verification and quality assurance. Execution speed and resource or monetary cost of running the model are important considerations for agents. Each model has different knowledge cut-of-date that might impact thinks like understanding of the latest technologies. Openness and hosting arrangements of the model have an impact on privacy, and pre-existing safeguards set limitations on what purpose the model can be used for. Trust towards the model creator and political reasons might contribute to model selection as well. While the open source model available for hosting locally are getting closer or even exceeding the performance of proprietary models in certain tests like high school math, the proprietary models are still outperforming open source models in most of the tests (OpenAI, 2025a; Vellum, 2025).

Prompting. Right prompting is often seen as the primary method for improving agents' performance, and numerous prompting best practices and guides have been published for different purposes (Černý, 2024; Cowan et al., 2023; Ekin, 2023; MacCallum & Lee, 2025; J. Wei et al., 2023). Although a good prompt is specific for the purpose, a synthesis of the mentioned guides provides a description of characteristics for a good prompt.

- A clear task, role, and tone instruct LLMs on what the human operator wants from it.
- The right temperature and guidance for hallucination. LLM needs to be told what to do when it doesn't know the answer, or make it ask for more instructions or data instead of guessing the answer.
- Examples and few-shot prompts steer LLM to understand what is needed from it and enable in-context learning.
- Structured output improves the possibility of chaining and further develops the result. For example, asking LLM to provide an answer in a table with specific column headers ensures that LLM completes all the fields.
- Delimiting and clearly distinct parts of the prompt from each other, making the instruction clearer and preventing certain types of prompt injections.
- Specified step-by-step instructions and Chain-of-Thought prompting where LLM is asked to "think" before it gives the final answer. Ask LLM to provide the reason why it ended up in that response.
- Persistence within agentic conversations. It is important to make the LLM understand that it should wait for other agents until all the conditions are satisfied before returning to the human operator.

Context and tools. Context engineering increases focus and speed of the agent by writing, selecting, compressing, and isolating just the right context for the agent (LangChain, 2025). In addition to instructing a specific focus on a prompt, external information can be provided for LLM by using the RAG technique. Using methods like Model Context Protocol (MCP), agents can invoke tool calls, bringing them capabilities like a calculator, clock, databases, and web search, and enabling them to communicate with other agents (Gan & Sun, 2025). Context window optimization, one of the key component of context engineering (Pajo, 2025), becomes relevant when conversation history and external data exceed build-in limits of the model, resource consumption restrictions, or expected response times. As the RQ1 results demonstrate, manually injecting context simply by attaching only the relevant data into the system or user prompt increases the performance of the agent while keeping the context window optimized.

Architecture. Simplicity and linearity improve long-running agents' performance and minimize out-of-sync related context problems, where agents don't know what other agents in the network are doing (Yan, 2025). Parallelism is needed when agents need highly specialized roles or are initiated based on various events in the operating environment. A multi-agent approach enables agents to have specific instructions and use an optimal LLM model for the task, which improves the collective performance of the agent network. Sharing of context and situational awareness can be done using centralized components. Also, parallel agent operations have their use cases, for example, creating multiple COAs and then rank each other's responses.

Safeguards. AI adversarial attacks and control bypassing tactics have been studied extensively (Greshake et al., 2023; Qi et al., 2023; Wan et al., 2023; A. Wei et al., 2023; Zou et al., 2023) and legislations like EU AI ACT mandate adversarial testing performed for high-risk AI systems (European Union, 2024). While tools like PyRIT (Microsoft AI Red Team, 2024) and SPLX (SPLX, 2025) provide a comprehensive set of automated tests, at least the following safeguards should be implemented.

- Input query moderation, as end users or other agents can control the user prompt.
- Implementing delimiters into the system prompt provides safeguards against prompt injections.
- Output response moderation prevents outbound lateral movement and data leaks.

Comprehensive implementation of the safeguards above is difficult. Agents' access to secret data, its exposure to untrusted content, and its ability to communicate externally form a lethal trifecta, which should be mitigated by eliminating one of the three elements (Willison, 2025).

3.2.3 Agent architecture construction

Essential components of a DCO AI agent are introduced in Figure 4. Human operator communicates with AI agent network using the user interface, storing and accessing documents from the datastore, reviewing and updating the status of tasks in the task list, and reviewing the operation status from event logs. Agent infrastructure consists of an agent framework orchestrating the whole agent network, reasoning and chat models used by agents to comprehend tasks, embedding models for converting documents in the datastore into vector representation, which can be understood by reasoning and chat models, and a vector database for storing and searching vectors. Several tools are used to offload task specific instructions from agents and to standardize activities like unified format event logging and embedding multi-format data into a vector database. Regardless of multiple components in the architecture, all the agent infrastructure can be hosted in a single computer, enabling operations in an isolated environment, for example, in military operations.

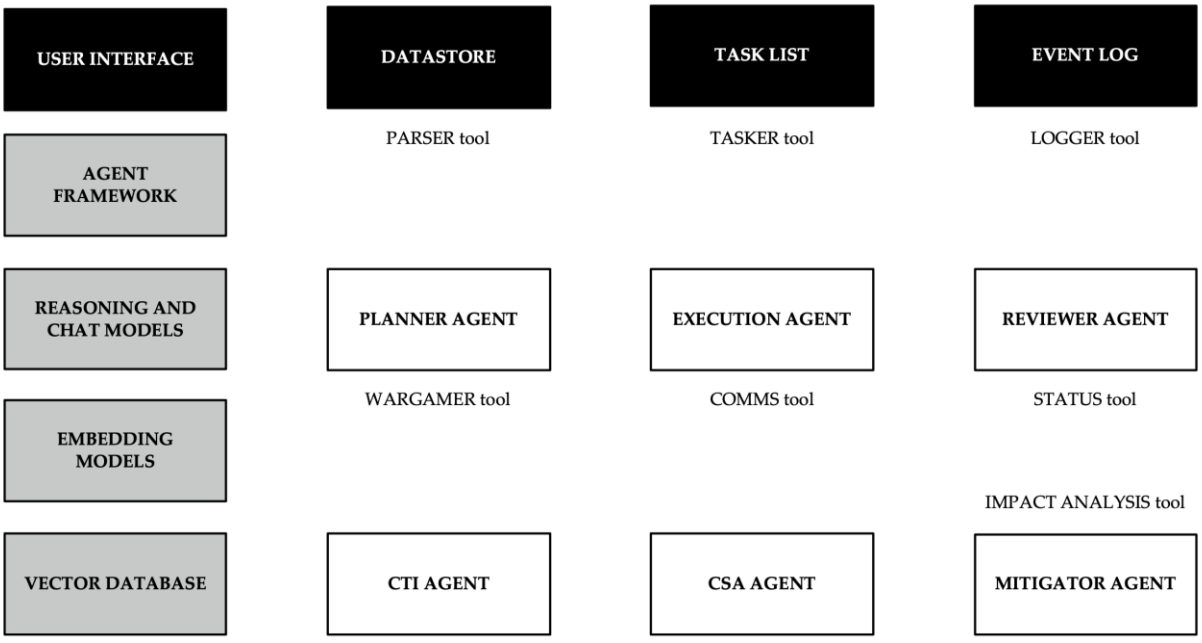

Figure 4: Essential components of DCO AI agent

The construction of the DCO AI agent architecture is introduced in Figure 5. In this architecture, each of the 6 agents has their specific role, while they share tools and resources, and are capable to communicate with each other.

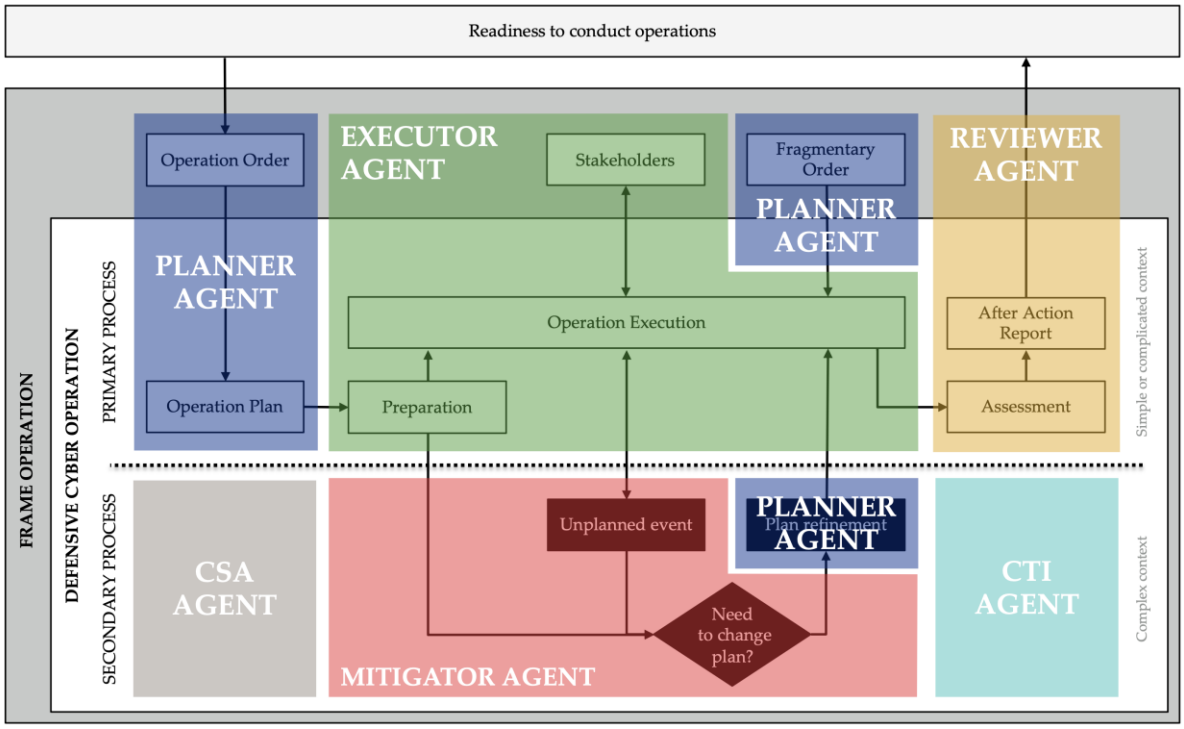

Figure 5: DCO AI architecture construction built on Sarjakivi et al (2025) DCO decision-making process

The *planner agent* is responsible of creating and updating the Operations Plan (OPLAN) based on higher headquarter issued Warning Order (WARNO), Operations Order (OPORD), or Fragmentary Order (FRAGO). Those orders are stored in the *datastore* and should include the objectives, area of operations, timeline, relation, and dependencies to frame operation, Rules of Engagement (ROE), and available support (Wade, 2023). As an agent, using a foundation model as its core, which is not pre-trained for a particular task, needs to understand the context where the operation is performed to create an actionable OPLAN fit for purpose. The *planner agent* uses general operations planning principles, like Military Decision-Making Process (MDMP) (Wade, 2023) and NATO cyberspace doctrine (NATO, 2020) as a base, but scenario and threat landscape, SOPs, specific doctrines, capabilities of the team, and lessons learned from previous operations needs to be context engineered into the agent. In addition to information included into prompts and found from the *vector database* and *event log*, the *planner agent* can call the *Cyber Threat Intelligence (CTI) agent* to understand the threat landscape and the *Cyber Situational Awareness (CSA)* agent to understand the current operational status impacting planning. *Planner agent* uses the *Wargamer tool* to challenge and improve its plans. *Wargamer tool* benchmarks the plan to alternative plans and plans from past operations and tries to find weaknesses from the plan by projecting the enemy responses to activities planned.

The *execution agent* is responsible for ensuring that the team enters the operations well-prepared and that the operation is executed smoothly. It divides OPLAN into tasks in the *task list* using the *tasker tool* and the *communications tool* to communicate with external stakeholders. Using the *status tool*, agent follows operation progress by comparing OPLAN to *event log* and task completion status and calls *CTI* and *CSA agents* for more information if needed. If part of the operation enters the secondary process in the DCO decision-making model, the *executor agent* calls the *mitigator agent* to bring the operation back to the primary process. Both *executor* and *mitigator agents* use the *impact analysis tool* to understand the impact of a cyber event on the overall operation. At the end of the operations, the *reviewer agent* assesses the operation based on the *event log* and the *status tool* and creates an AAR. AAR, together with identified lessons learned, are stored into the *datastore* and used to update SOPs, thus improving future operations performance.

4. Discussion

As a response to the national critical infrastructure's growing exposure to fast-evolving cyber threats, cyber defenders must improve their capabilities. This study demonstrated that AI agents can significantly accelerate decision-making in the most critical decisions DCO leaders need to make, namely, *prioritization of defended capabilities and assets*, and *the right resourcing and allocations.* Prioritization and generic plan generation scored high in the evaluation, and agents could generate out-of-the-box ideas and COAs quickly, supporting

human operators' innovation. While proper context engineering improves the quality of the agent's results, the HMT is needed as agents tend to hallucinate and provide varying results, for example, to fill all the cells in a table. Giving more details to the agent makes it focus on other details, leading to reduced focus on the original objective. Human oversight is needed, especially on resourcing-related tasks, as the agent didn't perform particularly well in that domain.

Even with rapid development with LLMs, AI agents are not ready for autonomously execute DCOs. Well-functioning HMT is needed to navigate a complex operating environment where COs are typically conducted. The asynchronous nature of cyber physical environments where humans and machines operate seem to be difficult for LLMs to comprehend.

This study provided a construction of a reference architecture for the DCO planning agent. Future research can be built to improve the results of the reference architecture. As AI's capabilities are developing rapidly, it is expected that the capabilities of autonomic AI agents will increase in the near future.

Acknowledgements

The authors would like to thank the Finnish Ministry of Defence for its support in writing this article (SOW275).

AI declaration: While producing this article, AI was used to improve the spelling and to perform a grammar check. As the studied target was an AI agent, the evaluated product naturally includes AI components.

Ethics declaration: No ethical clearance was needed for this study.

References

Azrilyant, J., Sidun, M., & Dolashvili, M. (2022) Fact and Fiction: Demystifying the Myth of the 85%, *The George Washington University Elliott School of International Affairs*.

Černý, J. (2024) Prompt Engineering: Tactics and Techniques in Open-Source Intelligence, *Journal of Information Warfare,* 23(3).

City of Saint Paul (2025) "Digital Security Incident Info Hub", [online], https://www.stpaul.gov/departments/emergency-management/digital-security-incident-info-hub.

Cowan, L. S., Lerman, D. C., Berdeaux, K. L., Prell, A. H., & Chen, N. (2023) A Decision-Making Tool for Evaluating and Selecting Prompting Strategies, *Behavior Analysis in Practice*, 16(2), 459–474.

DeepSeek. (2025) "DeepSeek-R1-0528 Release", [online], https://api-docs.deepseek.com/news/news250528.

Ekin, S. (2023) Prompt Engineering For ChatGPT: A Quick Guide To Techniques, Tips, And Best Practices. *Institute of Electrical and Electronics Engineers (IEEE)*.

European Union. (2024) Regulation (EU) 2024/1689 of the European Parliament and of the Council.

Fang, R., Bindu, R., Gupta, A., & Kang, D. (2024) LLM Agents can Autonomously Exploit One-day Vulnerabilities, arXiv preprint arXiv:2404.08144.

Federal Bureau of Investigation (2025) *Internet crime report 2024.*

Festor, P., Habli, I., Jia, Y., Gordon, A., Faisal, A. A., & Komorowski, M. (2021) Levels of Autonomy and Safety Assurance for AI-Based Clinical Decision Systems, *Computer Safety, Reliability, and Security* (Vol. 12853, pp. 291–296), Springer International Publishing.

Gan, T., & Sun, Q. (2025) RAG-MCP: Mitigating Prompt Bloat in LLM Tool Selection via Retrieval-Augmented Generation, arXiv preprint arXiv:2505.03275.

Goecks, V. G., & Waytowich, N. (2024) COA-GPT: Generative Pre-Trained Transformers for Accelerated Course of Action Development in Military Operations, *2024 International Conference on Military Communication and Information Systems (ICMCIS)*, 01–10.

Goecks, V. G., & Waytowich, N. R. (2023) DisasterResponseGPT: Large Language Models for Accelerated Plan of Action Development in Disaster Response Scenarios, arXiv preprint arXiv:2306.17271.

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., & Fritz, M. (2023) Not what you've signed up for: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection, arXiv preprint arXiv:2302.12173.

Hersey, P., Blanchard, K. H., & Johnson, D. E. (2013) *Management of organizational behavior: Leading human resources (10th edition)*, Pearson.

Johnson, M., & Bradshaw, J. M. (2021) How Interdependence Explains the World of Teamwork, *Engineering Artificially Intelligent Systems* (Vol. 13000, pp. 122–146), Springer International Publishing.

Kereopa-Yorke, B. (2024) ClausewitzGPT Framework: A New Frontier in Theoretical Large Language Model-Enhanced Information Operations, *Journal of Information Warfare*, 23(2).

LangChain. (2025) "Context Engineering", [online], https://blog.langchain.com/context-engineering-for-agents/.

MacCallum, N., & Lee, J. (2025) "OpenAI Cookbook: GPT-4.1 Prompting Guide", [online], https://cookbook.openai.com/examples/gpt4-1_prompting_guide.

Mbaiossoum, B. L., Mahamat, A. D., Batouma, N., Dionlar, L., Apollinaire, B. B., & Adam, I. O. (2025) How to Choose the Best AI LLM: A Guide to Navigating the Diversity of Models, *Journal of Information Systems Engineering and Management*, 10(34s), 221–232.

Microsoft AI Red Team. (2024) "Python Risk Identification Tool for generative AI (PyRIT)", [online], https://azure.github.io/PyRIT/.

NATO (2020) *Allied Joint Doctrine for Cyberspace Operations (AJP-3.20).*

NATO Cooperative Cyber Defence Centre of Excellence (2025) "Locked Shields 2025 Showcased Nations' Commitment to Defending Cyberspace", [online], https://ccdcoe.org/news/2025/locked-shields-2025-showcased-nations-commitment-to-defending-cyberspace/.

OpenAI (2025a) "Gpt-oss-120b & gpt-oss-20b Model Card", [online], https://openai.com/index/gpt-oss-model-card/.

OpenAI (2025b) "Introducing GPT-5", [online], https://openai.com/index/introducing-gpt-5/.

Pajo, P. (2025) Context Engineering: Enhancing Large Language Model Performance Through Comprehensive Contextual Management, Pre-print.

Qi, X., Huang, K., Panda, A., Henderson, P., Wang, M., & Mittal, P. (2023) Visual Adversarial Examples Jailbreak Aligned Large Language Models, arXiv preprint arXiv:2306.13213.

Sarjakivi, P. (2025) Artificial Intelligence Accelerated Cyber Operations: A Systematic Literature Review, *Journal of Information Warfare*, 24(1).

Sarjakivi, P., Ihanus, J., & Moilanen, P. (2025) Demonstration and Evaluation of Defensive Cyber Operations Decision-Making Model, *European Conference on Cyber Warfare and Security*, 24(1).

Smeets, M. (2022) The Role of Military Cyber Exercises: A Case Study of Locked Shields, *14th International Conference on Cyber Conflict*, 700, 9–25.

Snow, J. J. (2015) Entering the matrix: The challenge of regulating Radical Leveling Technologies*, Monterey, California: Naval Postgraduate School*.

SPLX (2025), "About Us", [online], https://splx.ai/about-us.

Vellum (2025) "LLM Leaderboard", [online], https://www.vellum.ai/llm-leaderboard.

Wade, N. M. (2023) *BSS7: The Battle Staff SMARTbook (7th ed.)*, The Lightning Press.

Wan, A., Wallace, E., Shen, S., & Klein, D. (2023) Poisoning Language Models During Instruction Tuning, arXiv preprint arXiv:2305.00944.

Wei, A., Haghtalab, N., & Steinhardt, J. (2023) Jailbroken: How Does LLM Safety Training Fail?, arXiv preprint arXiv:2307.02483.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2023) Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, arXiv preprint arXiv:2201.11903.

Willison, S. (2025) "The lethal trifecta for AI agents: Private data, untrusted content, and external communication", [online], https://simonwillison.net/2025/Jun/16/the-lethal-trifecta/.

Yan, W. (2025) "Don't Build Multi-Agents, Cognition", [online], https://cognition.ai/blog/dont-build-multi-agents.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023) ReAct: Synergizing Reasoning and Acting in Language Models, arXiv preprint arXiv:2210.03629.

Yin, R. K. (2003) *Case study research: Design and methods (3rd ed)*, Sage Publications.

Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., & Fredrikson, M. (2023) Universal and Transferable Adversarial Attacks on Aligned Language Models, arXiv preprint arXiv:2307.15043.