Human–AI Collaboration in Air Defense

From Training to Operational Missions

Johan Källström

Advanced Programs, SAAB

Agenda

- Human-AI Collaboration in Training
- Human-AI Collaboration in Operational Missions

Reinforcement Learning for Improved Utility of Simulation-Based Training

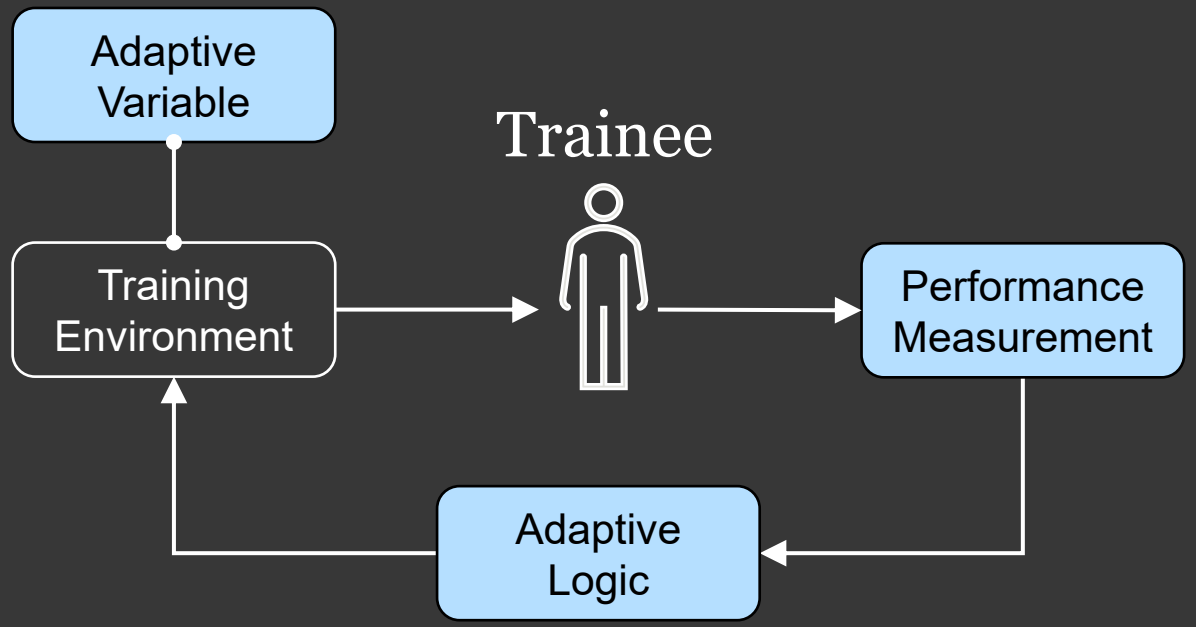
Human-AI Collaboration in Training

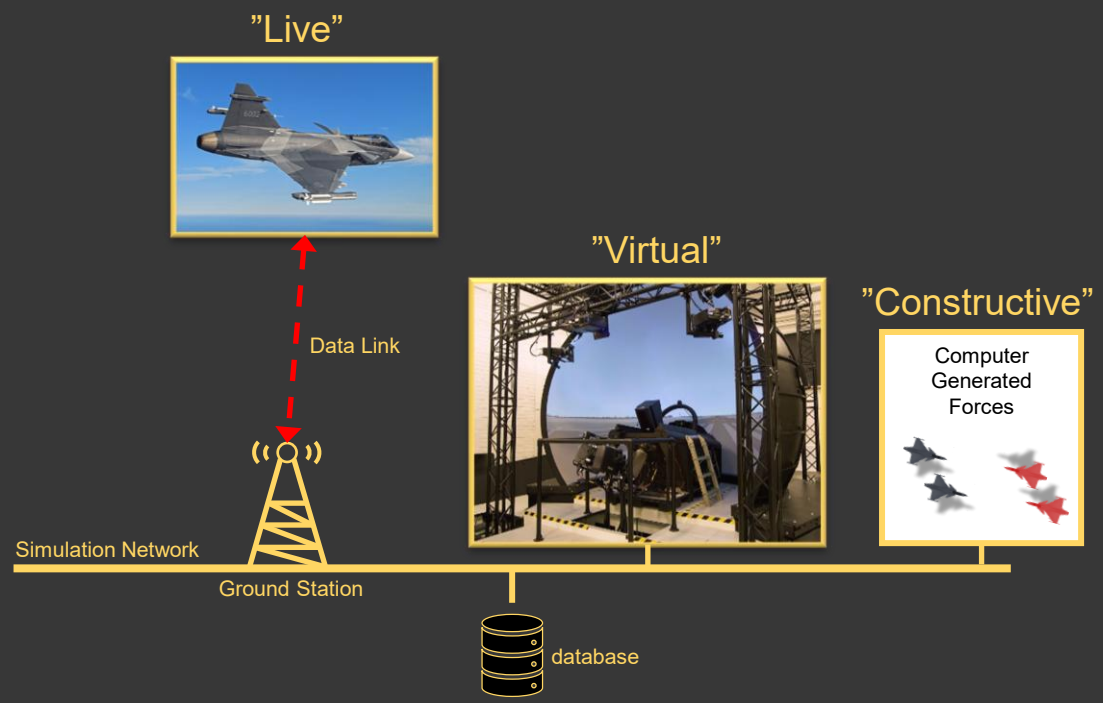
Johan Källström | HAICAD | Issue 1

WASP | WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

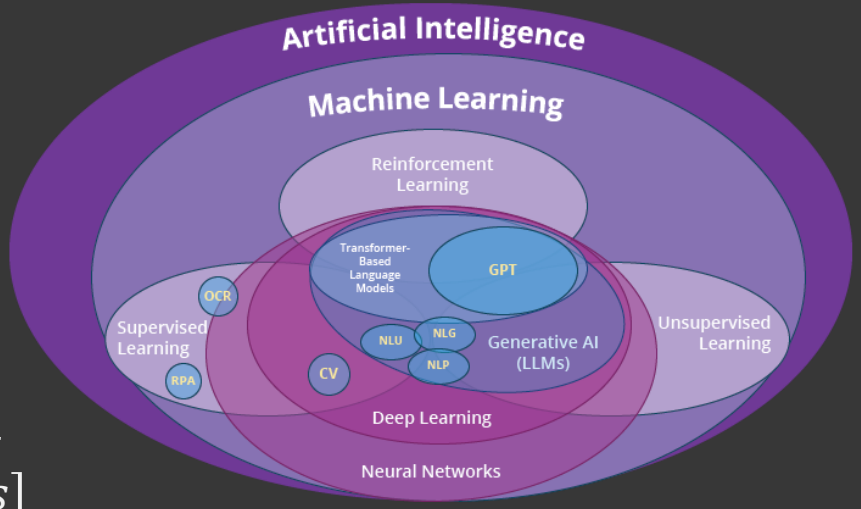
Research Goal

Use modern AI to construct adaptive pilot training systems

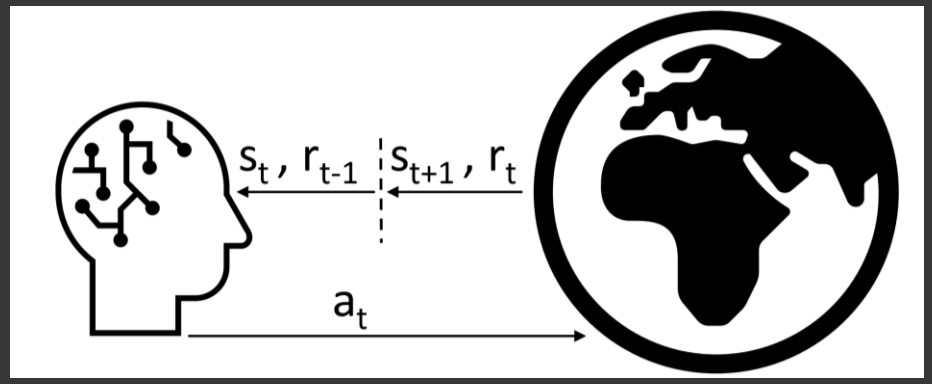




AI for Simulation-Based Training
Reinforcement Learning



- Learning through interaction with the environment, MDP = $\langle S, A, T, R, \gamma \rangle$
  - Represent goals with reward function $r_t = R(s_t, a_t, s_{t+1})$
  - Explore effects of actions, e.g. selecting actions by random
  - Reinforce high-value actions to improve policy
- Maximize future expected return G_t of policy π when starting in state s_t
  - Expected state value: $V^{\pi}(s) = E[G_t | s_t = s] = E[\sum_{k=0}^{\infty} \gamma^k r_{t+k} | s_t = s]$
  - Expected action value: $Q^{\pi}(s,a) = E[G_t | s_t = s, a_t = a] = E[\sum_{k=0}^{\infty} \gamma^k r_{t+k} | s_t = s, a_t = a]$
- Synthetic teammates & opponents are instructors in disguise and should optimize training value

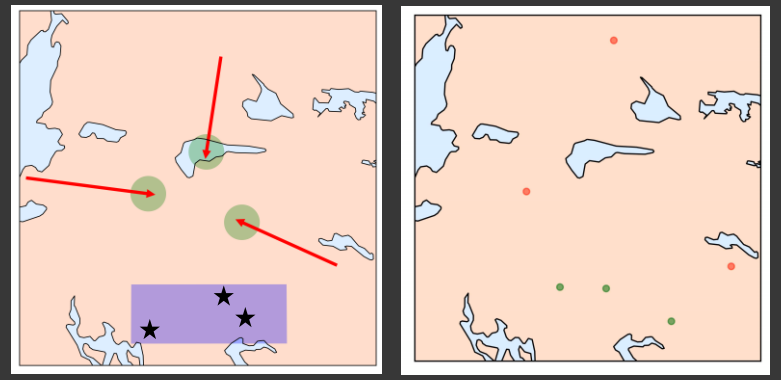


Cooperative Decision-Making

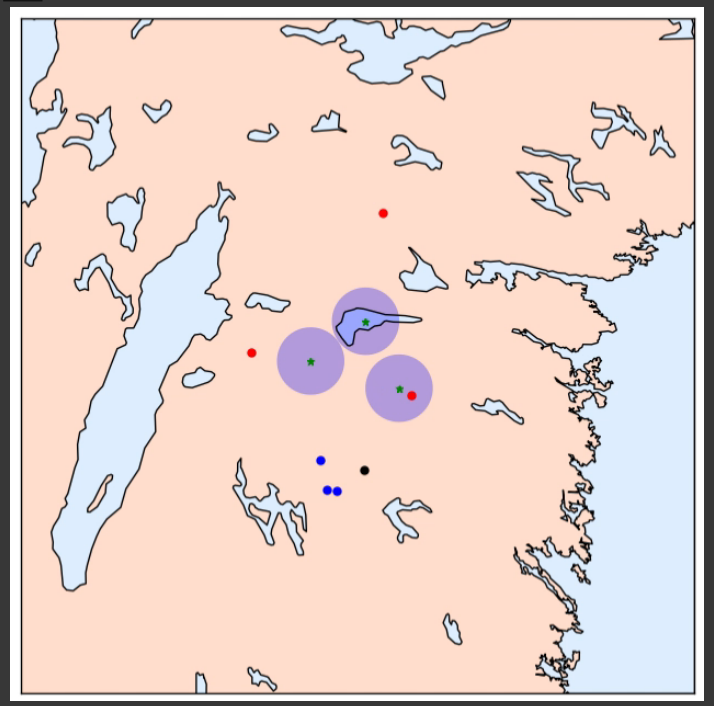
Defensive Counter Air

Blue agents aim to escort (by staying close to) Red out of protected air space

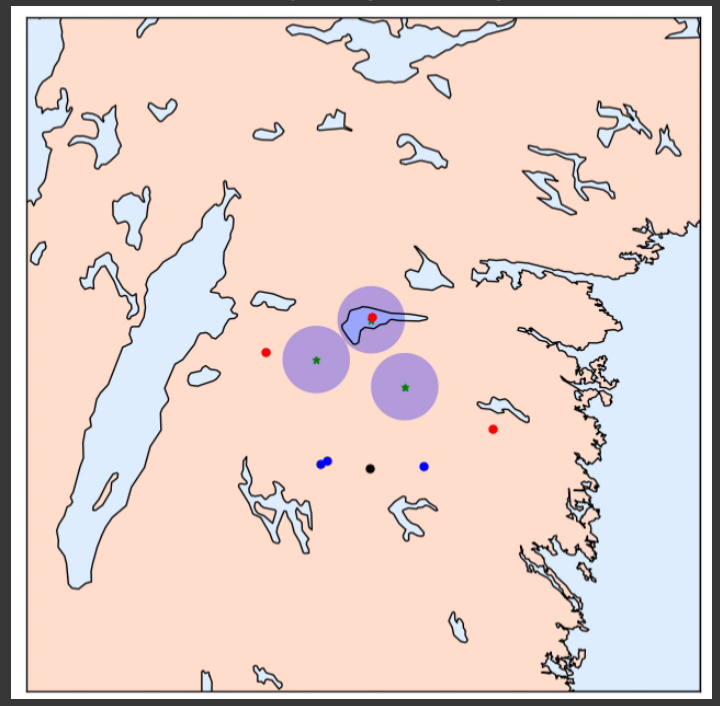
MDP
Obs: Relative position of other agents
Act: Turns with varying load factor [+coms]
Rew: Penalty for being far from any attacker
Transition function: CGF Flight simulator

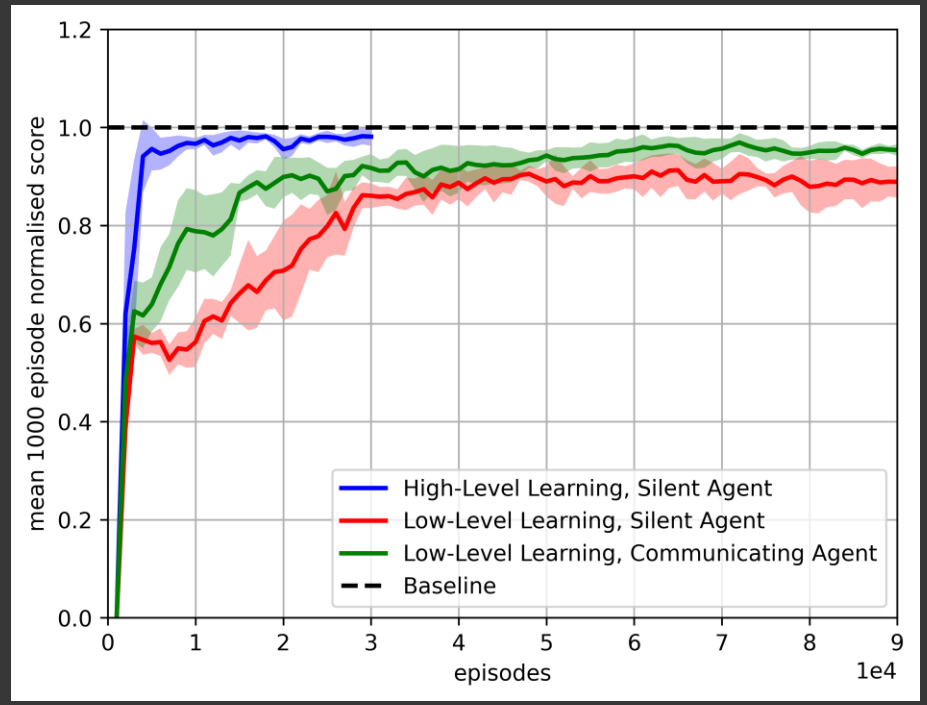


Learn navigation and target assignment
+ COM Channel for Learned Protocol

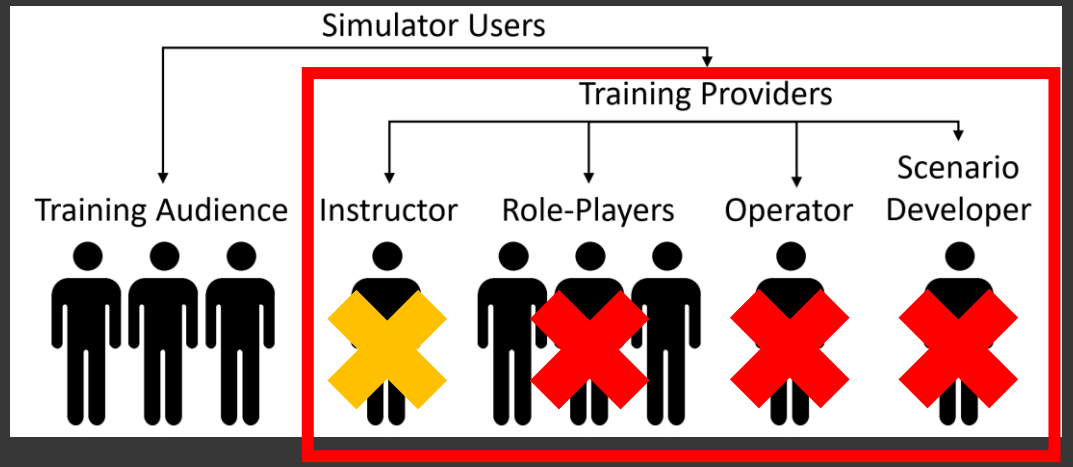


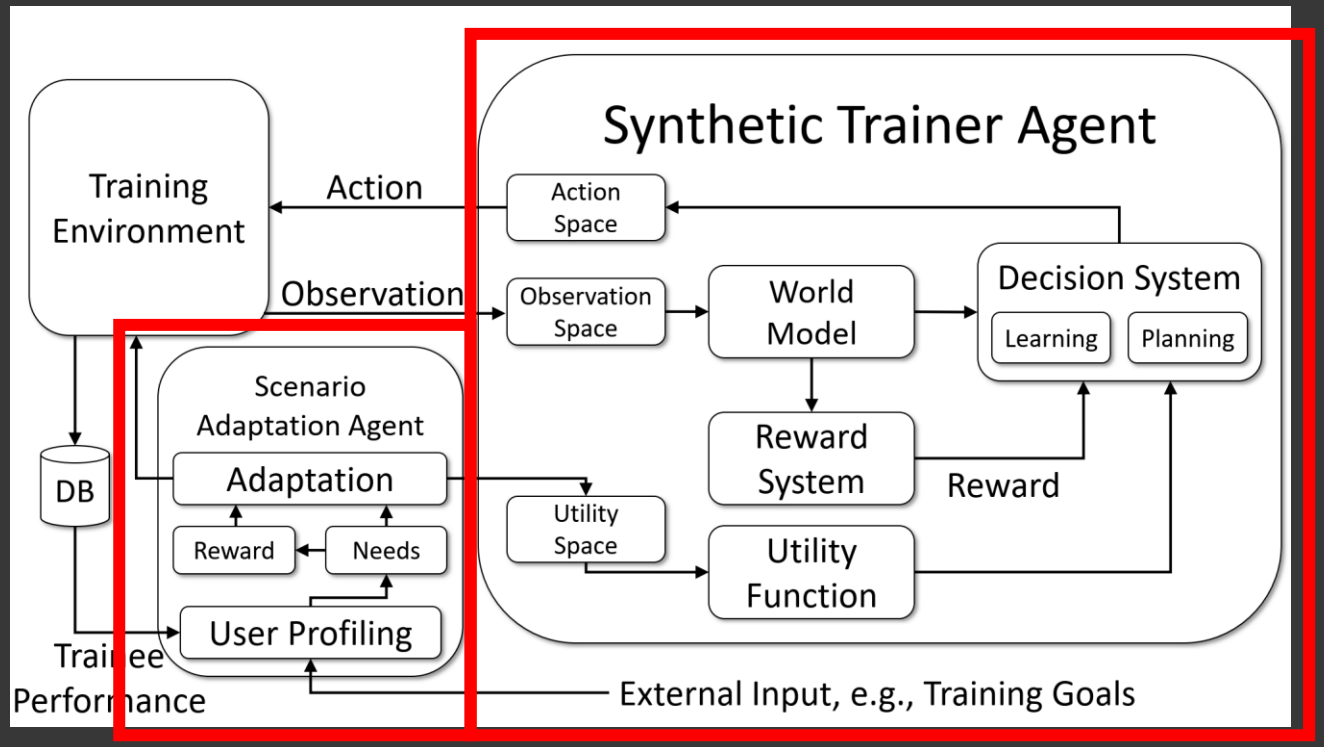
Learn only target assignment





System Concept





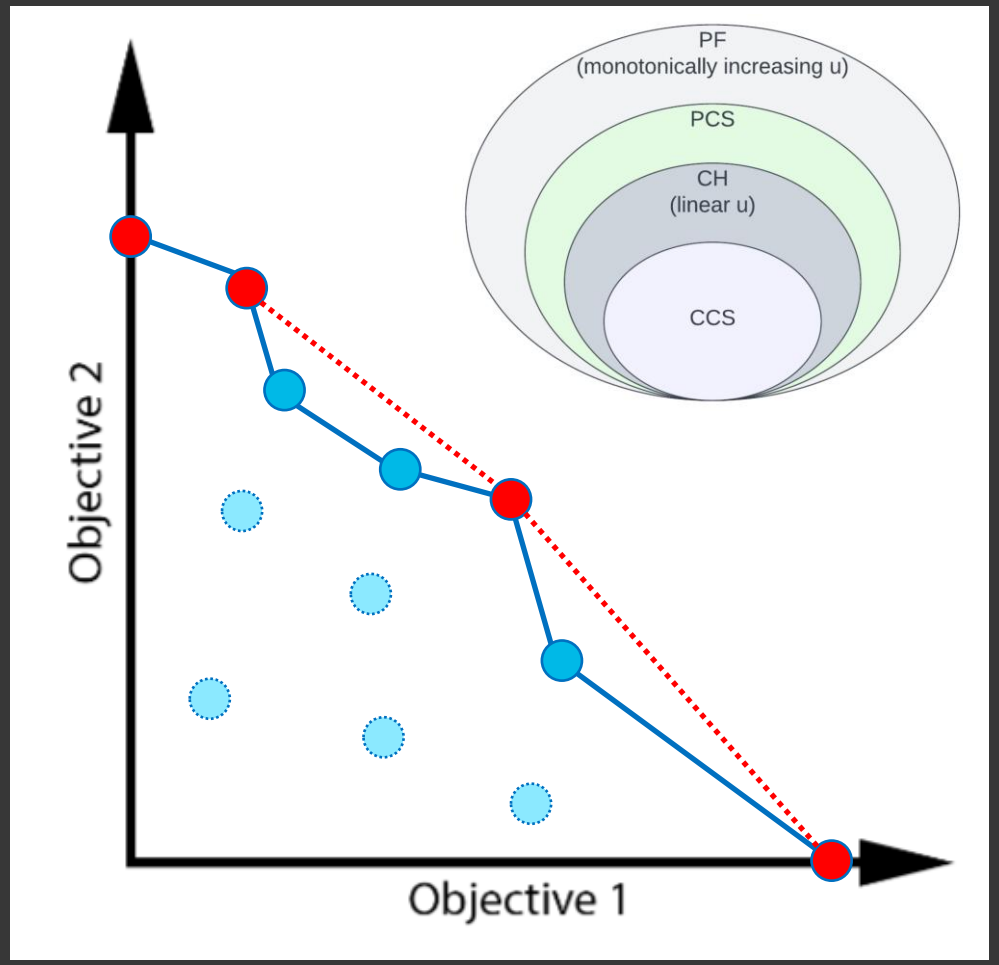
1. **Källström, J.**, Granlund, R., & Heintz, F. (2022). Design of Simulation-Based Pilot Training Systems using Machine Learning Agents. *The Aeronautical Journal*, *126*(1300).

Multi-Objective Reinforcement Learning

- Generalisation of RL to handle multiple conflicting objectives[2]
- Reward is a vector, with elements for each objective
- Optimal solution is a set of policies (e.g. **Pareto set/front**)
- Select policy according to user's **utility** function u

$$V_u^{\pi}(s) = u\left(E\left[\sum_{k=0}^{\infty}\gamma^k \boldsymbol{r}_{t+k} | s_t = s\right]\right), for\ SER\ criterion$$

$$V_u^{\pi}(s) = E\left[u\left(\sum_{k=0}^{\infty}\gamma^k \boldsymbol{r}_{t+k}\right) | s_t = s\right], for\ ESR\ criterion$$



2. Hayes, C. F., Rădulescu, R., Bargiacchi, E., **Källström, J.**, et al. (2022). A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems*, 36(1), 26.

Reward and Utility Functions

- Reward function[3]

$$\begin{bmatrix} TacticalGoals \\ ResourceConsumption \\ RiskExposure \\ \dots \end{bmatrix}$$

- Utility functions
  - **Weighted sum** (linear)

$$u = w_1 \times n_{opponents\_defeated} - w_2 \times n_{friends\_defeated}$$

E.g.: *I care 80% about surviving and 20% about defeating opponents*

  - **Threshold lexicographic ordering** (non-linear)

$u = n_{opponents\_defeated}$, while $p_{friends\_survival} > x$ and $p_{collateral\_damage} < y$

E.g.: *I should only try to defeat opponents if all my constraints are fulfilled*

3. Vamplew, P., Smith, B. J., **Källström, J.**, et al. (2022). Scalar reward is not enough: A response to Silver, Singh, Precup and Sutton (2021). *Autonomous Agents and Multi-Agent Systems*, 36(2), 41.

Multi-Agent Settings

In multi-agent settings, agents can have different rewards and utilities

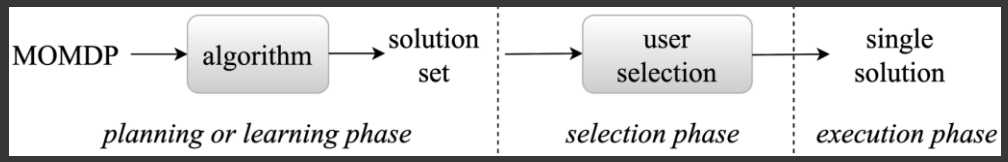
- Need to find trade-offs within a team
- Need to find best response to opponent's strategy

Solution concepts:

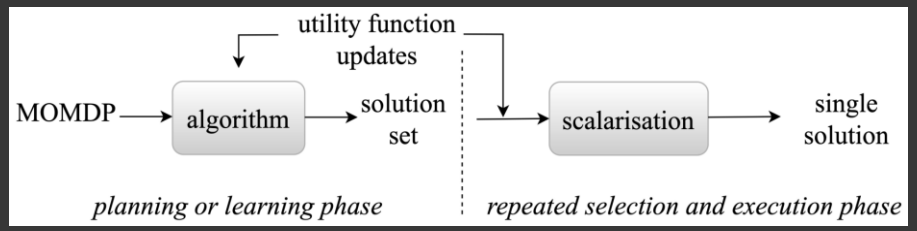
Reward	Utility Function: Team	Utility Function: Social Choice	Utility Function: Individual
Team	Coverage Sets	Mechanism design	Equilibria
Individual		Mechanism design	Equilibria

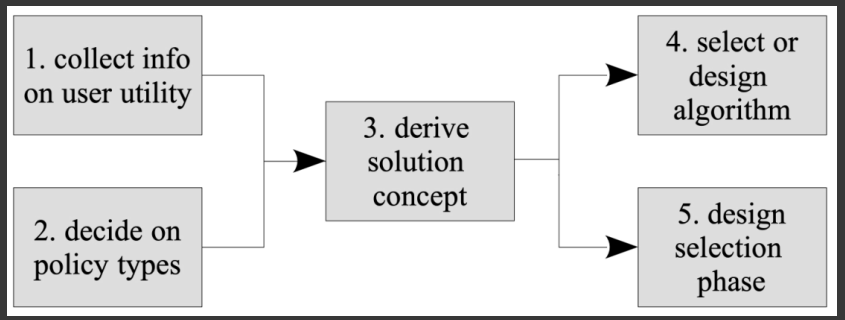
Use Cases

- Decision support – present policy set for selection



- Efficient policy adaptation to dynamic utility function through transfer learning





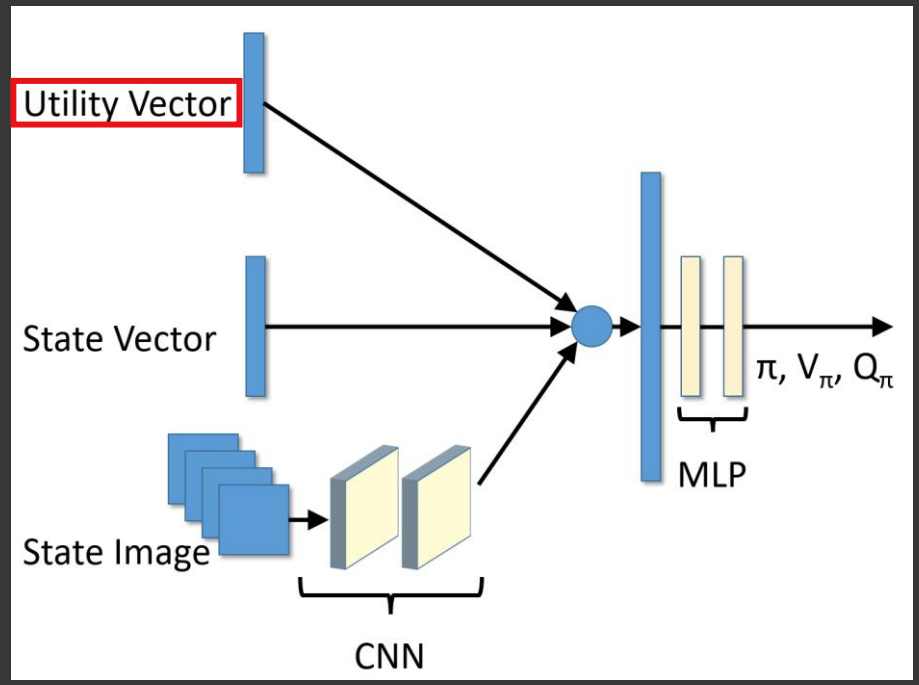
Utility-Based Approach[4]

4. Peter Vamplew, Cameron Foale, Conor F Hayes, Patrick Mannion, Enda Howley, Richard Dazeley, Scott Johnson, **Källström, J.**, et al. (2024). Utility-Based Reinforcement Learning: Unifying Single-objective and Multi-objective Reinforcement Learning. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems* (AAMAS)

Example: Provide Decision Support

Tunable Actor (T-Actor)[5]

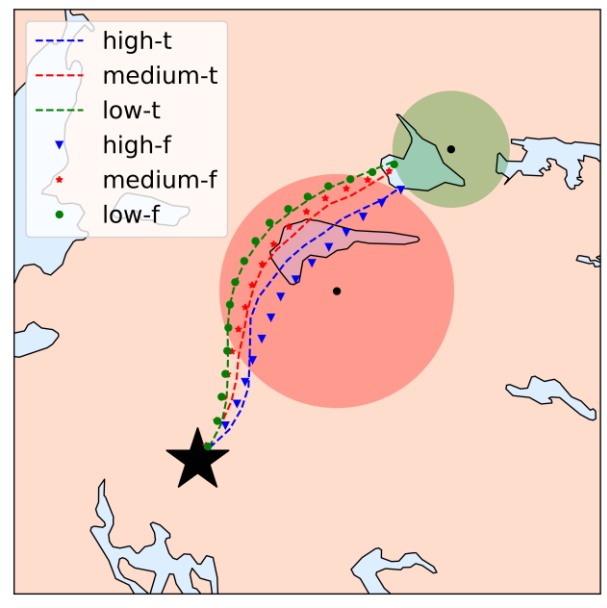
- Learns a set of Pareto optimal policies
- Policies represented by a single neural network
- Adapt behavior by specifying utility function, $a_t = \pi(s_t, \boldsymbol{u}_t)$
  - Learn with multiple utility functions
  - Maximize utility for each utility function
- Behavior can be modified after training



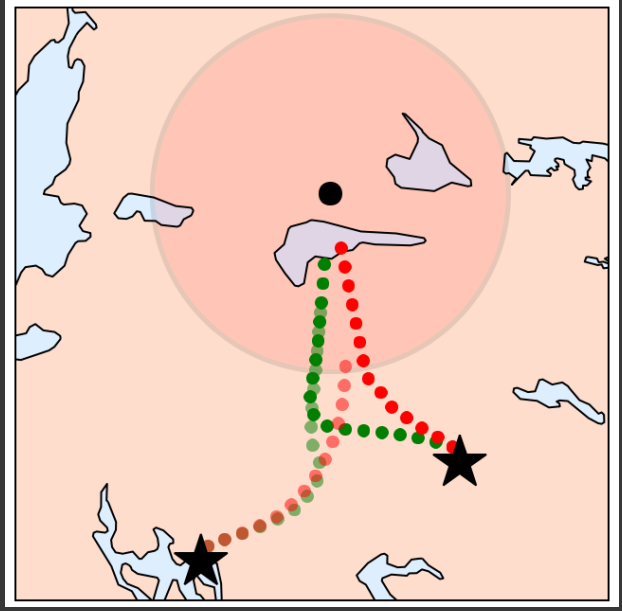
5. **Källström, J.**, & Heintz, F. (2019). Tunable Dynamics in Agent-Based Simulation using Multi-Objective Reinforcement Learning. In *Proceedings of the Adaptive and Learning Agents Workshop* (ALA) held at AAMAS.

Experimental Evaluation

Risk-aware navigation:
Balance time against risk exposure of air defence system

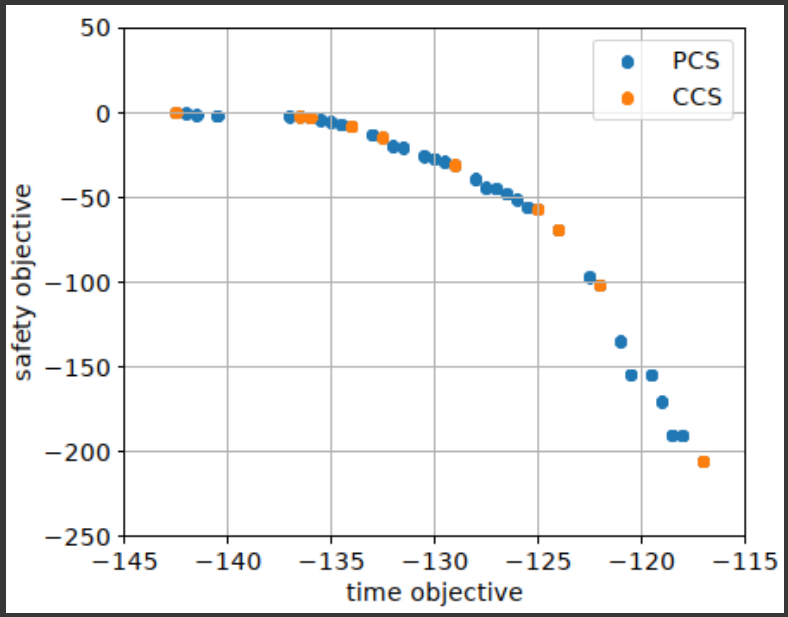


Utility Conditioned DQN[6]



Utility Conditioned MADDPG[7]

36 policies learned in one training run

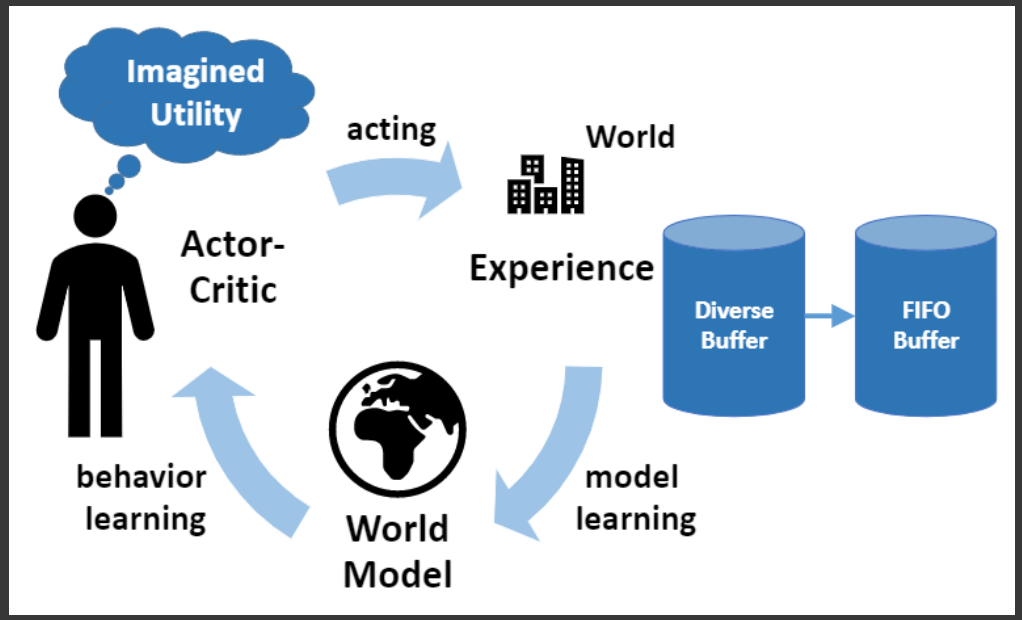


6. **Källström, J.**, & Heintz, F. (2019). Multi-Agent Multi-Objective Deep Reinforcement Learning for Efficient and Effective Pilot Training. *In Proceedings of the 10th Aerospace Technology Congress (FT2019)*.
7. **Källström, J.**, & Heintz, F. (2020, October). Agent Coordination in Air Combat Simulation using Multi-Agent Deep Reinforcement Learning. *In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*.

Example: Adapt to Dynamic Utility Functions

Model-Based Multi-Objective Reinforcement Learning[8,9]

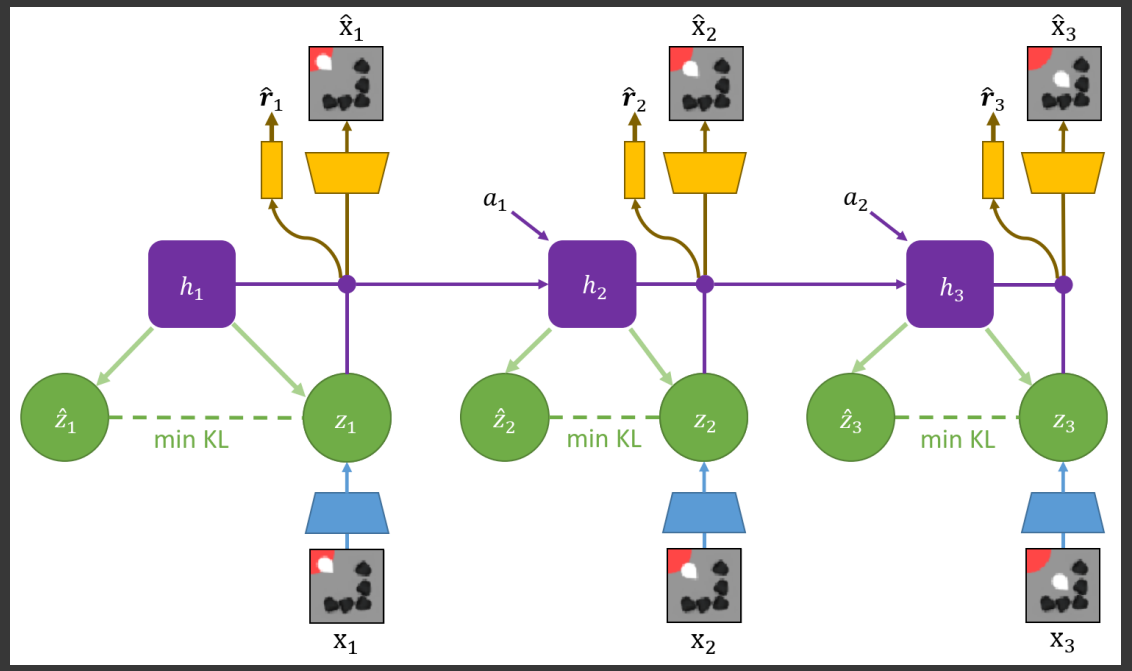
- Learns a MOMDP of the environment
  - World model based on DreamerV2 RSSM
- Explores with diverse utility functions in model
  - Prepare for changes in utility function



8. **Källström, J.**, & Heintz, F. (2023). Model-Based Actor-Critic for Multi-Objective Reinforcement Learning with Dynamic Utility Functions. *In proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems* (AAMAS).
9. **Källström, J.**, & Heintz, F. (2023). Model-Based Multi-Objective Reinforcement Learning with Dynamic Utility Functions. *In Proceedings of the Adaptive and Learning Agents Workshop* (ALA) held at AAMAS.

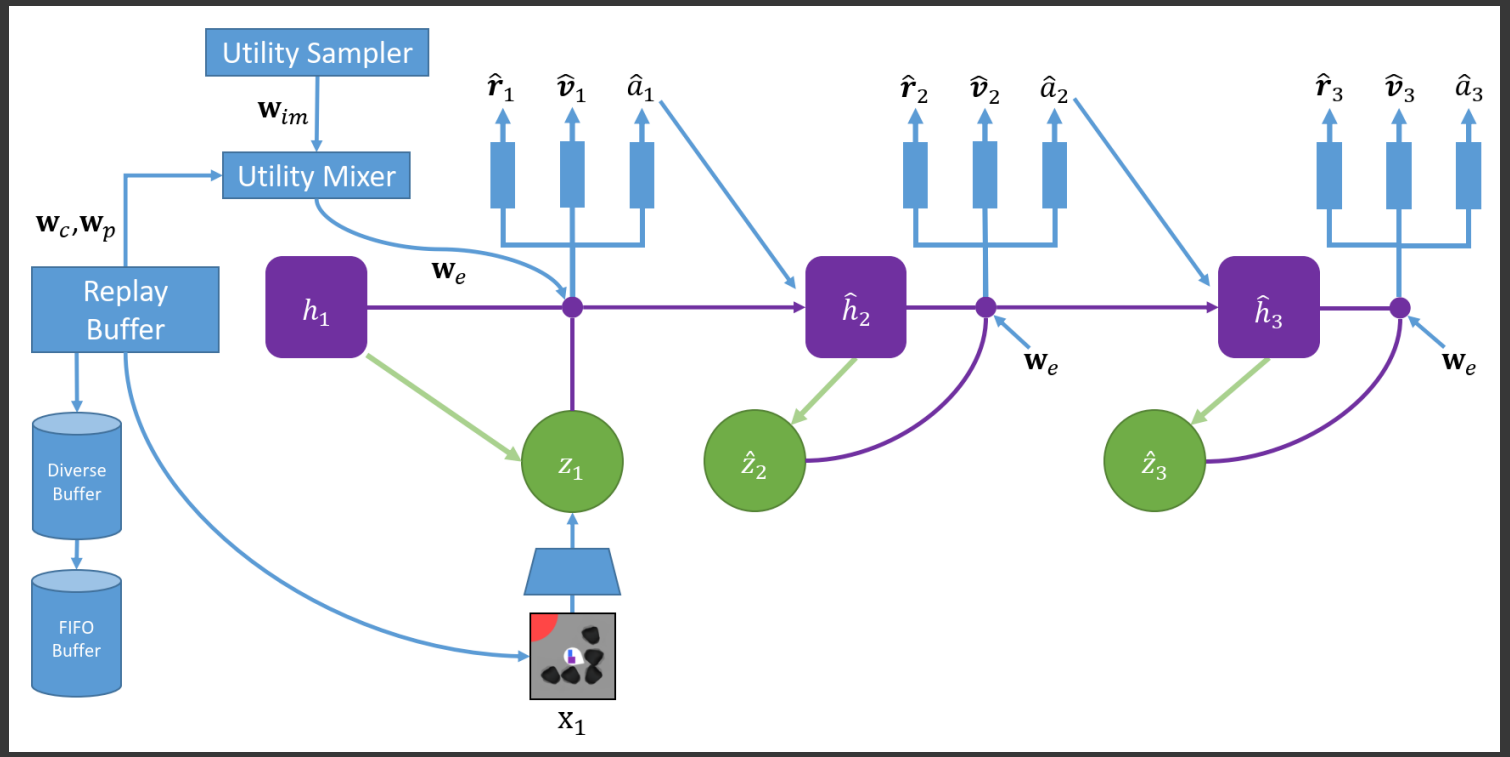
Multi-Objective World Model

- Multi-objective reward: $\hat{\mathbf{r}}_t \sim p_\phi(\hat{\mathbf{r}}_t | h_t, z_t)$
- Reward components as univariate Gaussians, gives loss contribution: $-\sum_{i=1}^{n} \ln p_\phi(r_t^i | h_t, z_t)$
- Diversity in replay buffer is enforced based on crowding distance; prioritise diverse sampling



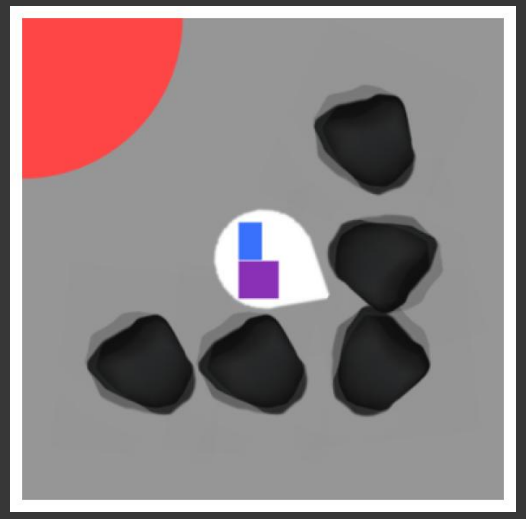
Imagination Rollouts

- Imagination rollouts use current ($\mathbf{w}_c$), past ($\mathbf{w}_p$) and imagined ($\mathbf{w}_{im}$) utility functions/weights
- Utility conditioned actor-critic learns in model
  - Actor: $\hat{a}_t \sim p_\psi(\hat{a}_t|\hat{z}_t, \boldsymbol{w})$
  - Critic: $v_\xi(\hat{z}_t, \boldsymbol{w}) \approx E_{p_\phi p_\psi}\left[\sum_{\tau \geq t} \hat{\gamma}^{\tau-t} \hat{\boldsymbol{r}}_\tau | \boldsymbol{w}\right]$

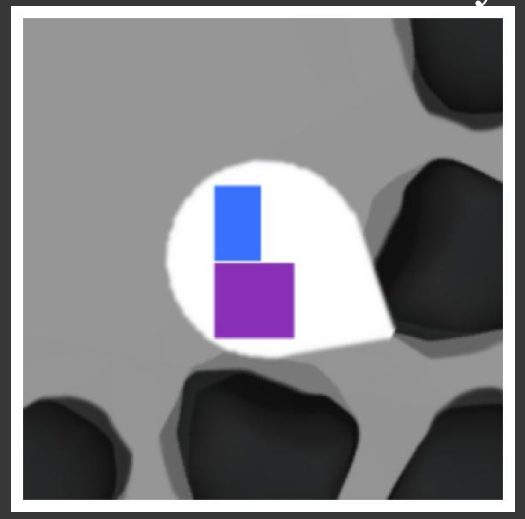


Experimental Evaluation

Minecart

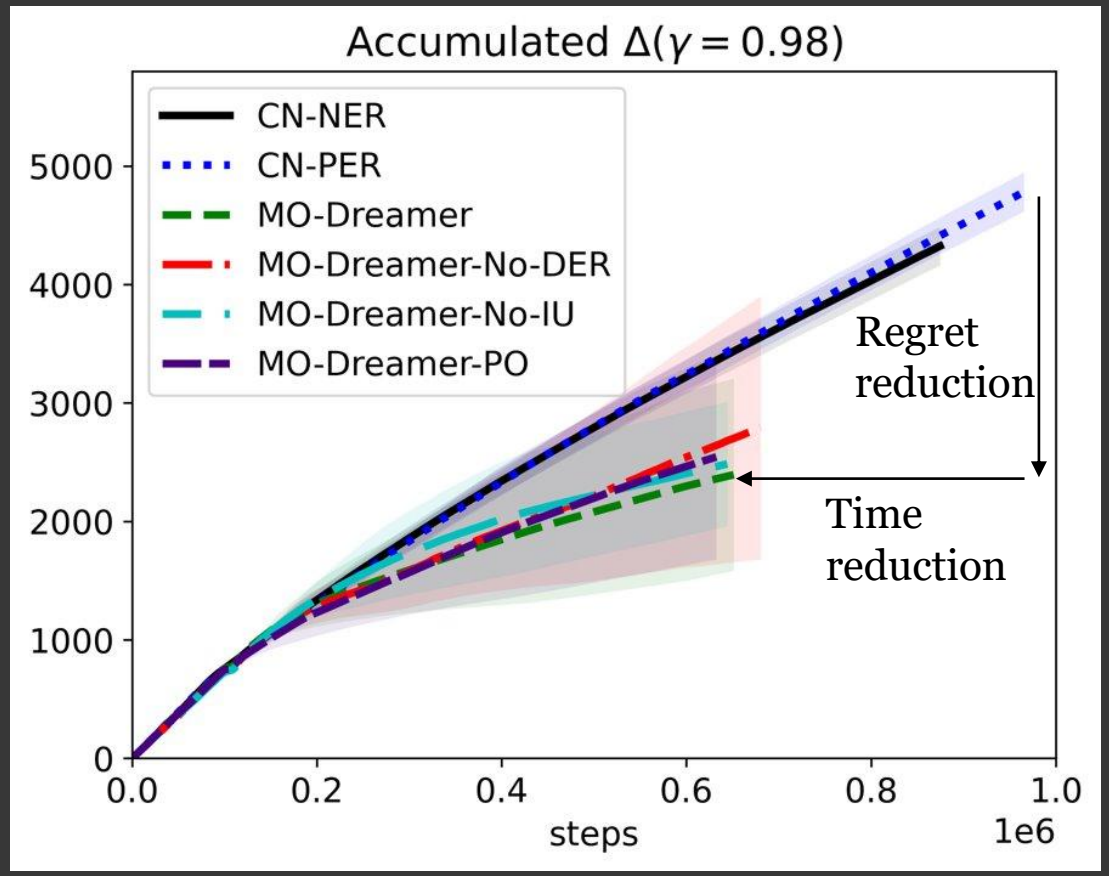


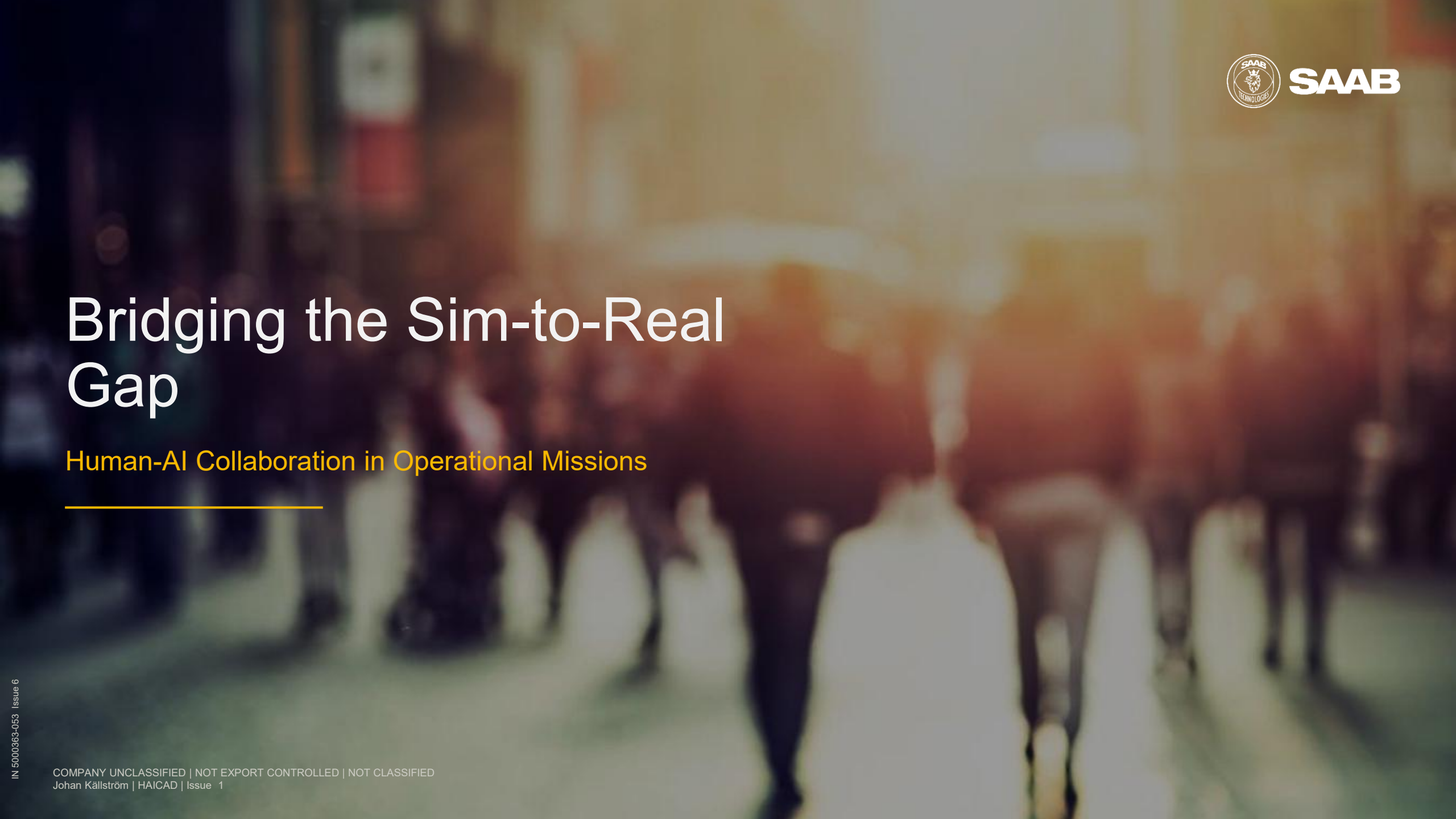
Partial Observability



- Outperforms human heuristic on Minecart

Δ = Regret = Optimal Value – Achieved Value
Less is better





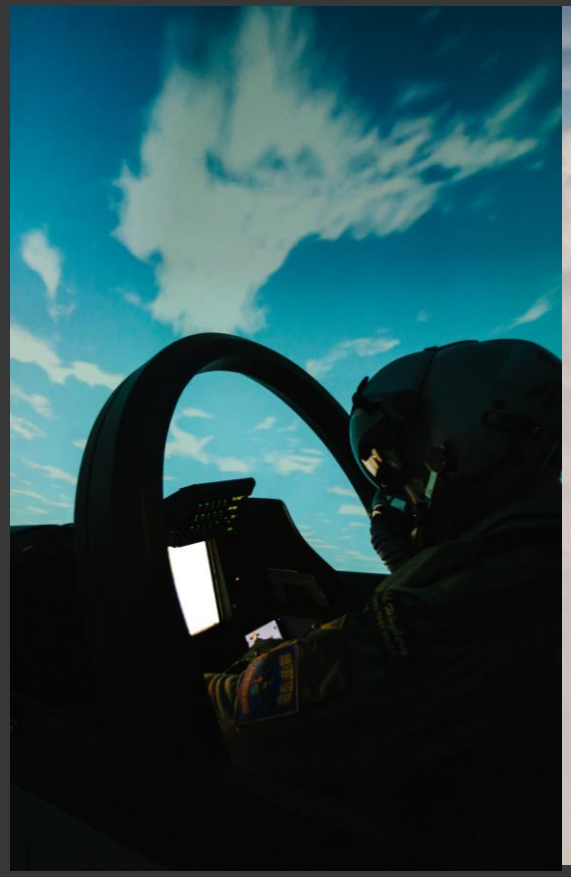
Bridging the Sim-to-Real Gap

Human-AI Collaboration in Operational Missions

Johan Källström | HAICAD | Issue 1

Project Beyond
In short

- Gripen E with AI vs Gripen E in air combat
- Accomplished - Milestones
  - AI-agent beating humans in simulator
  - AI-agent vs human in real flight
- Done in partnership with Helsing



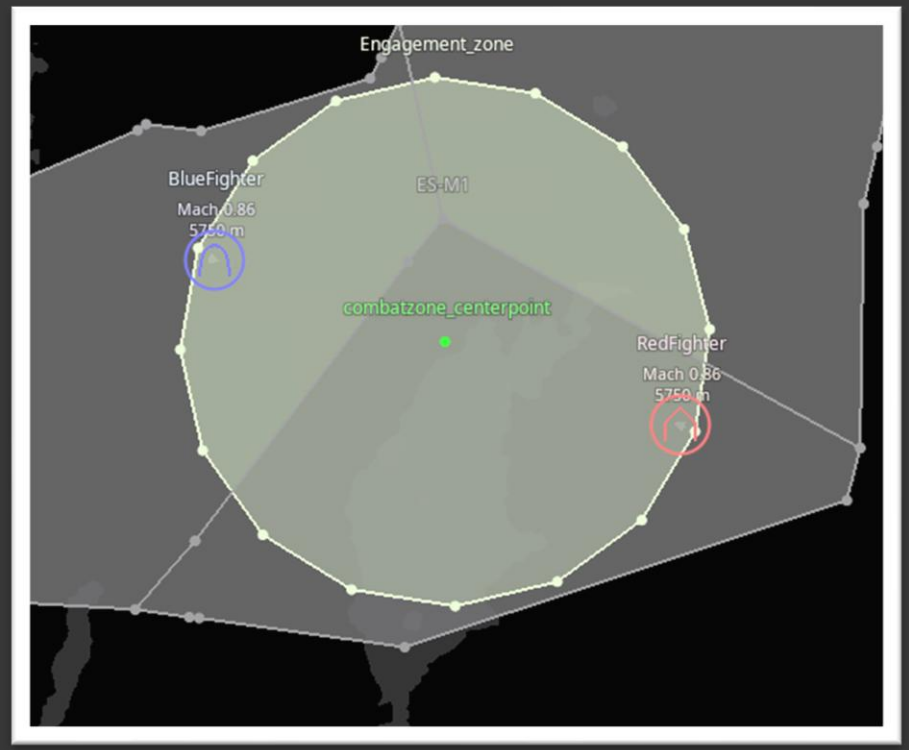
Project Beyond
Main scenario

Beyond Visual Range (BVR) air combat between two aircraft

In more detail:

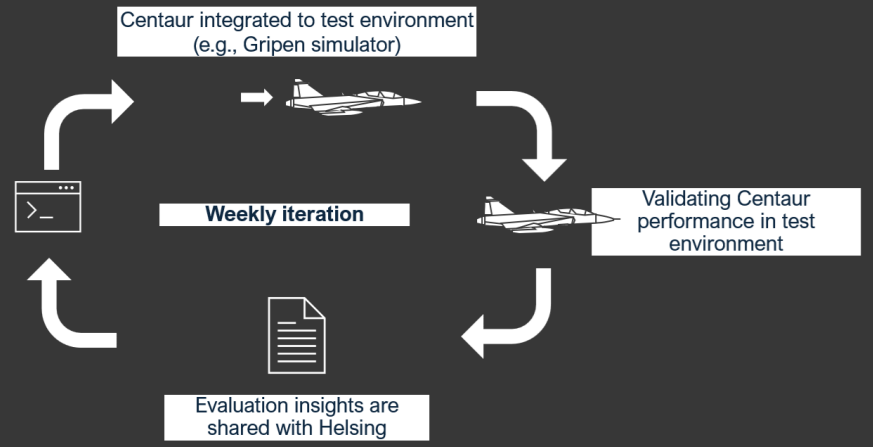
- Aircraft need to fly within a predefined combat zone
- Each aircraft is supported by external sensors (outside of the combat area)

The objective for each aircraft is to win the fight, if possible

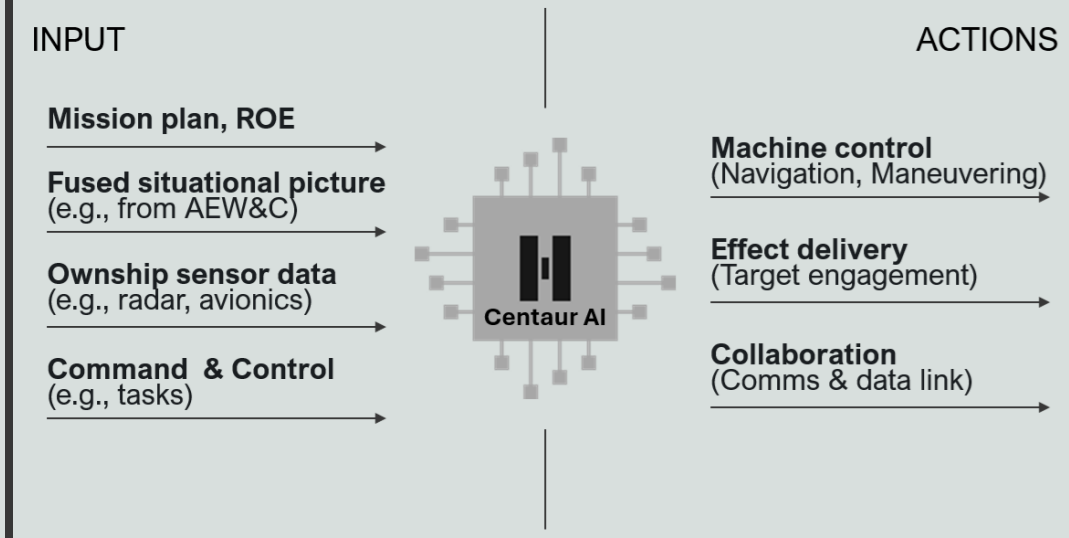


Project Beyond
Helsing's Centaur Agent

- Project done in partnership with Helsing
- Delivery of 1 improved agent per week
  - Training over 3-4 days
  - Gained Experience: ~30 years of virtual training time per agent across multiple instances
  - All experience is cumulated to one new agent version for delivery to Saab



- State-of-the-art AI agent based on Reinforcement Learning
- Developed by Helsing since 2023 – Software, Infrastructure, Tooling
- Optimized for complex air combat missions for current and next generation of combat air platforms (crewed/uncrewed)

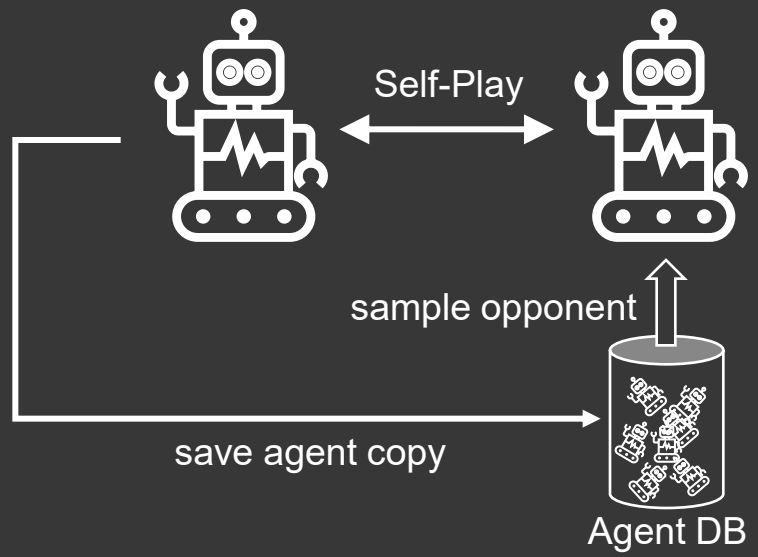


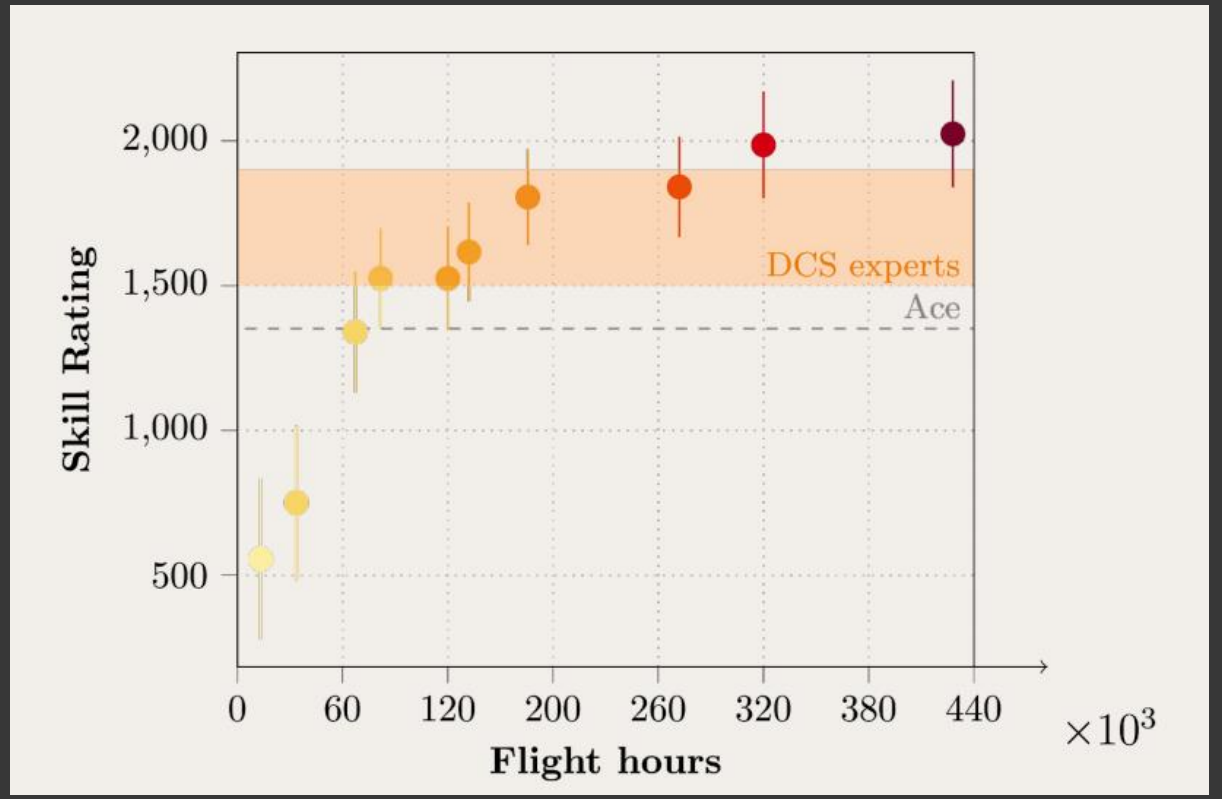
Johan Källström | HAICAD | Issue 1

Skill Improvement by Flight Hour

Reinforcement learning through self-play

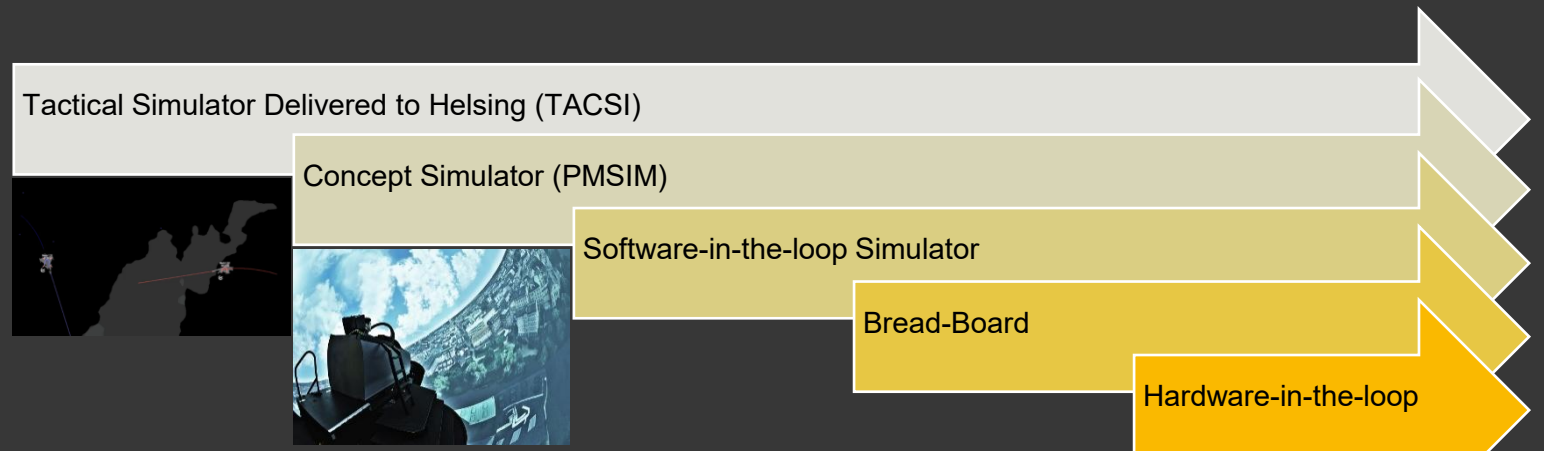
- Agents learn by interacting with copies of themselves
- Results in a form of automated curriculum with successively more challenging opponents

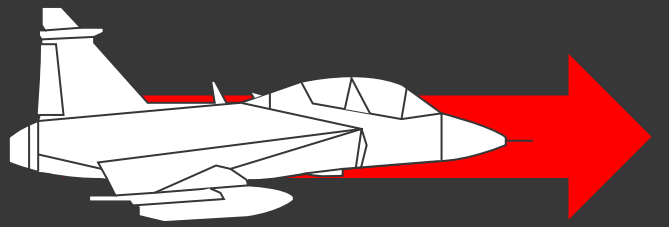




Sim-to-Real

- Increasing level of fidelity of simulator used for training and verification of the behavior model





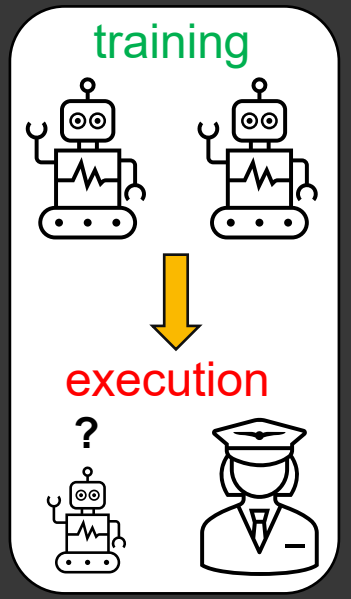
KICK-OFF *”Learning to fly”* *“Testing and permit-to-fly”* **LIFT-OFF**

1st of January | 15th of March | 9th of May | 28th of May

Training Agents for Robustness to Variations in the Environment

Moving from the simulation to the real world, the agent may experience changes in environment dynamics

- Platform dynamics
- Sensor and weapon performance
- Teammate and Opponent behaviour, including pairing of human and synthetic agents



These changes can be handled by training the agent with

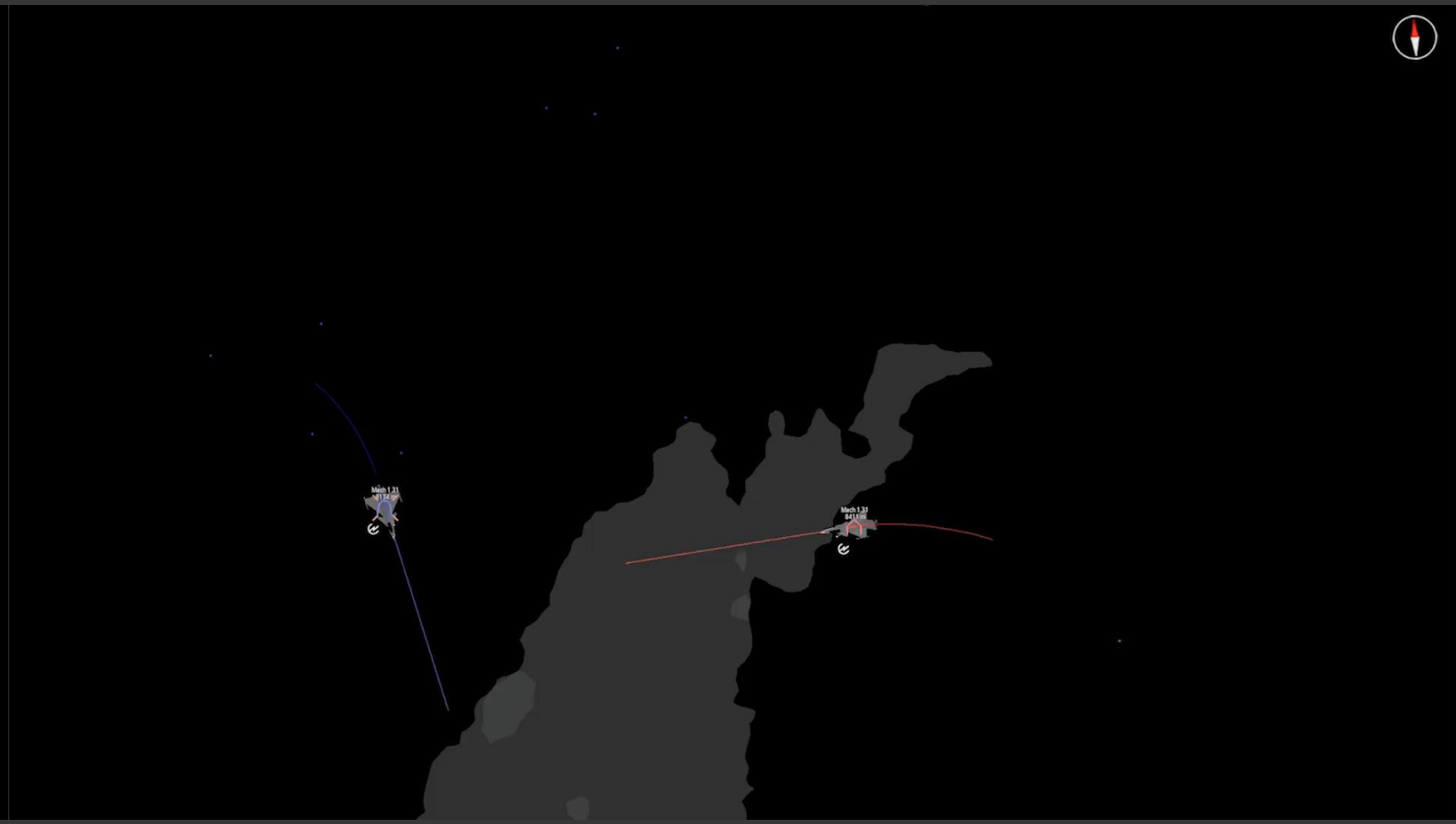
- variations in model parameters
- varitaions in opponent behaviour

Project Beyond

Beyond visual range (BVR) – Simulated – Human pilot vs AI-agent

Human pilot

AI-agent



Johan Källström | HAICAD | Issue 1

Conclusions

Working explicitly with multiple objectives in reinforcement learning can

- support non-expert users in finding suitable behavior models
- help adapt efficiently to changes in training needs or operational requirements
- provide more informative value functions

Lessons learned from Project Beyond

- Necessary to consider sim-to-real gap for good performance
- International collaborations in particular may increase the sim-to-real-gap due to challenges in sharing sensitive information
- Aircraft integration may also require retargeting of agent software and require additional testing

Human–AI Collaboration in Air Defense

Questions?

Johan Källström

Advanced Programs, SAAB