Estimation and Prediction for Advanced Situational Awareness

Louise Lennartsson*†, Zoran Sjanic*†, and Gustaf Hendeby*
* Dept. of Electrical Engineering, Linköping University, Linköping, Sweden
e-mail: firstname.lastname@liu.se
† Saab AB, Linköping, Sweden
e-mail: firstname.lastname@saabgroup.com

I. Background

Autonomous systems are revolutionizing the world with promises of more efficient, better, and safer operations, especially as humans are gradually taken out of the loop. High-level, high-quality *situational awareness* (SA) is key to this development. It is SA that allows humans, as well as autonomous systems, to understand the physical world around them and decide how to interact with it. Both current and, especially, future platforms depend on increasingly more advanced SA to handle different situations which these systems will face.

One application with high demands on SA is fighter jets, where the cognitive load of the pilot is high.

An important aspect of air-to-air combat is to have an information advantage over the opponent, *i.e.* build superior SA within a network of many (possibly heterogeneous) mobile platforms while trying to remain undetected by the opposition.

Typically, modern combat aircraft have at least one onboard sensor and communicate with other agents on a network to share information. Distributed calculations are preferable, *i.e.* each agent computes its own SA using the information available to it, since a decentralized approach is robust to the loss of agents.

A classic way to deny the opponent SA is by jamming to decrease the signal-to-noise ratio of the opponents' sensors. Historically the opponents' radars have been the target of jamming, but GPS jamming is becoming increasingly common. This not only impairs the opponent's ability to see objects in the surroundings, but also the ability to position their aircraft. Another way is to build the aircraft with special techniques and materials to lower its signature at different frequencies and make them harder to detect (*e.g.* minimizing *radar cross section* (RCS)). At the cost of own SA, remaining undetected can be achieved through conservative use of active sensors (*e.g.* radars) and communication channels, *i.e.*, one or several members of the team are avoiding use of active sensors.

These things lead to a messy battlefield, where every piece of information must be used to gain advantage over the opponent.

This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP), funded by Knut and Alice Wallenberg Foundation.

II. Research challenges

The end goal is to develop methods for SA that can handle scenarios where several agents are tracking multiple targets simultaneously.

In air combat it is desirable that SA and target tracking is close to real time. However, many algorithms do not scale well with large scenarios. A key question will hence be: what approximations can be made to ensure acceptable quality in the SA while still maintaining (almost) real time capability?

Another question is, how can an agent know what information is the most beneficial to its neighbors? It is important to be conservative with communication both from a tactical standpoint to avoid detection but also to decrease the load on the communication network. Here, inspiration can be drawn from [2]. Another thing to be careful of in decentralized target tracking is data incest. If an agent receives information from neighbors, how can it be guaranteed this information is not used twice? This inherently difficult problem will be addressed by taking inspiration from (generalized) CI and robust optimization techniques [1, 4].

If an agent's own position is uncertain (from *e.g.* GPS jamming), how can it know the (absolute) position of a target it is tracking? How can it use information from neighbors and how to communicate useful information to them in this case? To tackle this problem, the Simultaneous Localization and Mapping (SLAM) algorithm [6] provides some theory and insights.

With the high maneuverability of fighter jets it is possible to trick the tracking filter such that it loses the track. An accurate motion model increases robustness to maneuvering targets and a good start is to look at multiple-model methods like the *interacting multiple model* (IMM) filters [3].

III. Problem Exploration

The solution will be based on *Poisson multi-Bernoulli mixture* (PMBM) filter theory [7] and be designed to be used to obtain large scale SA from several heterogeneous agents with different fields of view and position uncertainties and using advanced target models. This will be used first in a *large scale centralized SA* setting and then introduced to the more challenging problem of *large scale distributed SA*.

A scenario with one agent and three targets has been designed to start investigating the problem, see Fig. 1. The figure shows three targets traveling in straight lines (left to right and downwards) that cross each other at some point. The agent's radar measures the targets and also some background clutter. The background clutter, crossing target paths and missed detections are common in practice to make the scenario more realistic. Unfortunately they also increase the complexity of the tracking problem.

The scenario is created using Stone Soup [5]. Stone Soup is an open source Python framework for developing tracking algorithms, and it contains standard components that can be mixed and matched. Fig. 2 shows the result when tracking the targets using a global nearest neighbor (GNN) multi target tracker, based on an Extended Kalman filter (EKF).

This gives a centralized baseline which can be used to compare new methods to. With Stone Soup being open source existing algorithms can be modified or new ones can be implemented to fit the framework. A first step is to improve the baseline with a PMBM filter.

In a decentralized solution each agent solves their own SA problem based on own sensors and the information communicated to them. As the communication is a limited resource, the information communicated has to be prioritized in some way. By increasing the number of agents and targets in the scenario, computational bottlenecks can be identified. These can be investigated to see where approximations can be made to speed things up. A single agent's SA or the collection of all agents' SAs can be compared to the centralized solution.

To investigate which type of information has the most impact on the SA the information shared between agents can be varied. Depending on which information and how it is communicated there is risk of data incest. It will be investigated how big impact this has on the SA and what can be done to mitigate it.

By varying an agent's position uncertainty it can be determined how big impact this has on the resulting SA, both in the agent itself and also for neighbors it shares information with.

The motion model impacts how well a tracking algorithm manages to keep a track alive. A passenger plane and a fighter jet have very different behavior and capabilities. With the information of what is being tracked, the filter can be tuned to improve tracking for each target type. As a next step, it can be investigated if the type of the track can be determined while tracking, using additional information. Such information could be non-kinematic, such as RCS, or aggregated over time, *e.g.* by keeping in mind previous maneuvers.

References

[1] R. Forsling, A. Hansson, F. Gustafsson, Z. Sjanic, J. Löfberg, and G. Hendeby. Conservative linear unbiased estimation under artially known covariances. *IEEE Trans. Signal Process.*, 70:3123–3135, 2022.

[2] R. Forsling, F. Gustafsson, Z. Sjanic, and G. Hendeby. Decentralized data fusion of dimension-reduced estimates using local information only. In *2023 IEEE Aerospace Conference*, pages 1–10, Mar. 2023.

[3] T. Kirubarajan and Y. Bar-Shalom. Kalman filter versus IMM estimator: When do we need the latter? *IEEE Trans. Aerosp. Electron. Syst.*, 39(4): 1452–1457, Oct. 2003.

[4] T. Li, J. M. Corchado, and S. Sun. On generalized covariance intersection for distributed PHD filtering and a simple but better alternative. In *20th International Conference on Information Fusion Fusion*, July 2017.

[5] Stone Soup contributors. Stone soup 1.6 documentation, 2025. URL https://stonesoup.readthedocs.io/en/v1.6/index.html. Accessed: Mar. 2025.

[6] Y. Xia, E. Stenborg, J. Fu, and G. Hendeby. Bayesian simultaneous localization and multi-lane tracking using onboard sensors and a SD map. In *27th International Conference on Information Fusion Fusion*, 2024.

[7] X. Xie, Y. Wang, J. Guo, and R. Zhou. The multiple model Poisson multi-Bernoulli mixture filter for extended target tracking. *IEEE Sensors J.*, 23(13):14304–14314, July 2023.

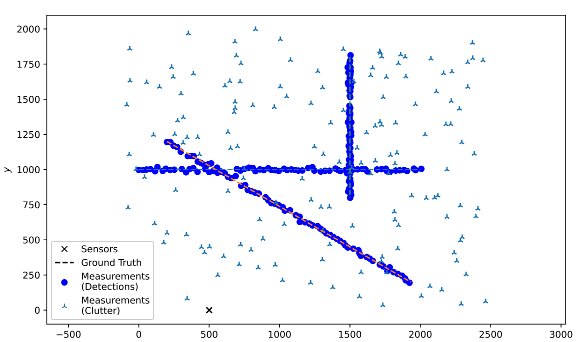

Fig. 1: A sensor measures three targets which generates both "true" detections and clutter.

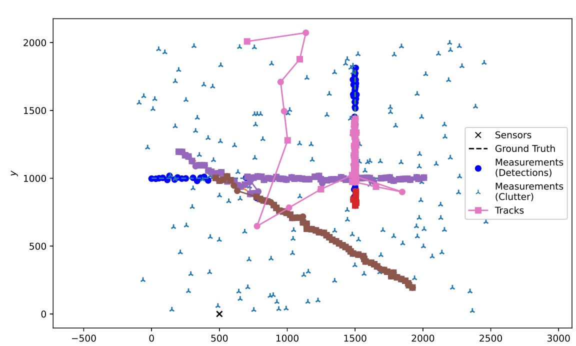

Fig. 2: Target tracking solution using GNN and EKF.