Multi-Stage Data Fusion and the MSTWG TNO Datasets

Stefano Coraluppi and Craig Carthel
Applied research Department
NATO Undersea Research Centre
Viale S. Bartolomeo 400, 19126 La Spezia, Italy
coraluppi@nurc.nato.int, carthel@nurc.nato.int

Abstract—Data fusion is a branch of applied science that processes measurements from a variety of sources and time epochs to provide a consolidated state history of a reality of interest, be it a physical process, an intangible system (e.g., economic, social, ...), or a complex entity that includes elements of both (e.g., a set of individuals in physical space). As such, data fusion is an important component in military and security surveillance systems. The technology encompasses stochastic modeling, nonlinear filtering, data correlation, and sensor management. Thus, there are significant technical overlaps with the signal processing, automatic control, information theory, and operations research communities. This paper provides illustrations of some applications of data fusion technology to surveillance systems. The unifying theme of the examples is the use of innovative and flexible multi-stage fusion processing. Additionally, we explore the use of multi-stage processing in the context of some of the datasets from the Multistatic Tracking Working Group (MSTWG), and we describe recent work on performance evaluation of tracking systems.

Keywords: Multi-sensor target tracking, data fusion, distributed processing.

1 Introduction

Data fusion technology is essential to military surveillance systems. In a typical theatre of interest, a number of similar or complementary systems will be available, each with a distinct sensor footprint, active or passive modality, sensor revisit rate (in the case of an active system), calibration and measurement errors, detection performance characteristics, etc. Sensor signal processing will typically yield detection-level (contact-level) returns; in some cases, pre-detection data is available. Some sensor systems, particularly older legacy systems, will lack important information such as contact localization error characteristics and measurement time information. Further, the data is often collected in a distributed network of sensors, so that bandwidth constraints and data latencies will hamper the quality, quantity, and timeliness of data available for sensor fusion.

Historically, military surveillance research has focused heavily on sensor technology. Downstream sensor fusion and target tracking technology has received less attention, and is an area where considerable performance gains remain to be achieved.

An overview of surveillance domains of interest, military platforms and systems, computational and communications constraints, and technical approaches to sensor fusion is beyond the scope of this work. We will focus here on providing a brief overview of fundamental paradigms for data fusion in section 2, with an emphasis on multi-hypothesis tracking. Section 3 introduces the unifying theme for the paper: high-performance tracking requires an effective choice of multi-stage data fusion architecture. Further, we describe multi-stage fusion architectures that in specific settings are shown to outperform single-stage, centralized, track-while-fuse processing. We denote these approaches as track-before-fuse, fuse-before-track, track-extract-track, and track-break-track. Section 4 discusses multi-stage processing in the context of the Multistatic Tracking Working Group (MSTWG) datasets provided by TNO, and section 5 discusses recent progress on robust metrics for tracker performance evaluation. Section 6 provides conclusions.

2 Multi-Hypothesis Tracking

An encyclopedic-style overview to approaches to data fusion is provided in [1]. Some approaches are appropriate for expeditionary operations that do not require real-time surveillance; as an example, area clearance prior to passage of a high-value unit requires surveillance results at the end of the data acquisition period. This allows for powerful batch-processing methods to be brought to bear on the problem [2]. On the other hand, scan-based methods must be utilized for real-time surveillance. Optimal data fusion remains a holy grail of sorts, in that all proposed fusion paradigms are known to invoke a number of simplifying algorithmic assumptions. The most powerful current approach to data fusion is multi-hypothesis tracking, which was first introduced in the late 1970s [3] and made feasible in the mid-1980s with the track-oriented approach [4]. A number of enhancements to the basic approach have appeared over the years [1].

Our contributions to data fusion technology span applications to ground, undersea, and maritime military surveillance systems. A central contribution has been the development of a computationally-efficient, high-
performance, and flexible multi-hypothesis tracking approach that enables multi-stage fusion processing: track-before-fuse in ground and undersea domains [5-6], fuse-before-track in large sensor fields [7], track-extract-track in the maritime domain [8], and track-break-track in difficult multi-target scenarios [9]. Our application of these techniques to challenging surveillance problems is ongoing, including non-military applications [10]. Examples drawn from these references will be presented here. Before doing so, we first illustrate the basic track-oriented multi-hypothesis tracking (MHT) approach with a simple example, shown in figure 1.

![Figure 1. A simple MHT example.](image)

The example assumes that two tracks, T1 and T2, have already been resolved. That is, prior data association decisions have led to a single global hypothesis that includes two tracks. Next, assume that a scan of data is received with two measurements, R1 and R2. Assume further that both R1 and R2 can feasibly be associated with T1, while only R1 can feasibly be associated with T2. This leads to a number of local (or track) hypotheses. Note that this set of hypotheses includes track continuation in the absence of a measurement (often denoted a track coast), as well as new-track hypotheses. A second scan of data includes a single measurement R3. We assume that R3 provides feasible updates to track hypotheses that include R2, as well as spawning a new-track hypothesis. Note that we assume that tracks are terminated after two coasts, indicated by the red icons in figure 1.

While the example includes a number of track hypotheses, it is important to note that each global hypothesis provide a compete set of data-association decisions that account for all resolved tracks and all sensor measurements. The power of the track-oriented approach is that we do not explicitly enumerate global hypotheses.

Each track hypothesis has an associated log-likelihood score that reflects track initiation and termination penalties as well as nonlinear filtering scoring; in the case of linear Gaussian systems, this scoring is based on the filter innovations [11]. The vector $c$ includes the track-hypothesis scores. We are interested in the optimal global hypothesis, which amounts to identifying a vector $x$ such that the global log-likelihood is maximized: the maximum likelihood solution. Having identified this solution through a two-stage relaxation approach based on linear programming or Lagrangian relaxation [12] (solution is noted in yellow in figure 1), many conflicting local hypotheses are removed. In particular, those track hypotheses that differ in the first scan past the resolved hypothesis layer are removed, while those that differ in the more recent past are maintained.

Having pruned the set of track hypothesis trees (with 5 surviving track hypotheses), we are ready for a new scan of data. In the example, the resolved layer always lags the current time by one scan: thus we have a multi-hypothesis example with hypothesis-tree depth ($n$-scan) of one.

### 3 Multi-Stage Data Fusion with the NURC DMHT

Multi-stage fusion with the NURC DMHT has two defining characteristics that differ from many legacy systems that exist today [1]. The first is that each tracker module retains measurement-level information at the output. That each, each module performs the following: it removes large numbers of measurement data, and associates the remaining measurements to form tracks over time. If the tracker is working well and the data is of reasonable quality, false measurements will be largely removed, and target-originated measurements will be largely maintained, and associated into tracks that persist over time with limited fragmentation. Since measurement data is available at the tracker output, optimal track fusion and state estimation is achievable in downstream tracker modules; the cost to achieve this performance benefit is a slightly larger bandwidth requirement between processing stages. The second defining characteristic of the NURC DMHT is that the track fusion is achieved in real time, with a scan-based approach. Traditionally, track fusion is performed in a post-processing batch mode that is not readily amenable to real-time surveillance application [1].

Figure 2 illustrates a multi-stage architecture for two platforms, with two monostatic source-receiver combinations, and two bistatic combinations. Data from each source-receiver combination is fed to a distinct tracking module, with subsequent track fusion.

![Figure 2. Example multi-stage data fusion architecture.](image)
The theoretical optimality of unified, batch and centralized approaches to fusion and tracking (track-while-fuse) is at odds with a number of practical considerations. First, in many surveillance settings optimal processing algorithms are either not known, or are computationally infeasible. Second, detection-level data may not be available from some propriety or legacy sensor systems; thus, in general it may be required to process a mix of track-level and measurement-level data. Finally, as we will see in subsequent sections, improved performance can be achieved with multi-stage processing that involves simpler and less computationally intensive algorithms than with near-optimal centralized processing. The DMHT provides an ideal tool to explore the superior performance that can be achieved with distributed, multi-stage algorithms.

3.1 Track-before-fuse

The first multi-stage architecture of interest is track-before-fuse (as illustrated in figure 2). In ground target tracking applications, where complimentary multi-sensor and multi-timescale data is available, tracking of high-rate data followed by fusion with informative low-rate data provides good performance results [5]. In undersea surveillance, where detection data exhibits significant target fading effects (i.e. highly correlated detection-event sequences), single-sensor tracking followed by real-time multi-sensor fusion provides superior results to centralized tracking, particularly for modest to high detection thresholds [6]; given complimentary looks on the target, it is best for each source-receiver to exploit its own detection trends, with subsequent track fusion. Nonetheless, the track-before-fuse architecture must be used with caution: for low detection thresholds, the fading-target effect is overcome by the need for small association gates, consistent with the high rate of detection files processed in a single, centralized tracking stage. Sea trial examples of these competing effects are in [13].

3.2 Fuse-before-track

Recent years have seen a trend towards unmanned multi-sensor surveillance networks with large number of cheap and limited-performance sensors. While these networks hold great potential for surveillance performance, it is of interest to quantify fundamental performance limitations. When a large number of equal-performance sensors is available in a given surveillance region, the intuitive assumption that more is better breaks down. The performance-degradation phenomenon is due to the sub-optimal processing inherent in all known approaches to data fusion. In particular, scan-based approaches that process each frame of sensor data as it reaches the sensor data and provide a real-time surveillance output are particularly challenged in the large sensor field environment. Batch processing approaches perform better, but they do not provide a real-time surveillance output.

We have recently introduced a two-stage architecture, fuse-before-track (FbT), which exploits the benefits of both batch and scan-based processing [7]; further, improved performance of FbT processing over centralized (single-stage) tracking has been demonstrated with simulated sensor data. The FbT architecture is illustrated in figure 3.

An illustration drawn from the performance studies in [7] is given in figure 4. In particular, we see a close-up on one target in a larger simulation-based analysis. False contacts are shown as black dots; target-originated contacts are shown as magenta dots; target trajectories are shown in magenta, single-sensor tracks are red, multi-sensor tracks are blue, and FbT tracks are cyan; the (intermediate stage) contact fusion output in FbT processing is shown with cyan dots.

The figure illustrates what is shown quantitatively in [7]; namely, when faced with large number of synchronized sensors with coincident coverage (in the performance study, ten sensors are assumed), it is best to combine measurement scans through a static fusion.
operation that leverages more powerful batch processing techniques than what can be achieved with scan-based processing. Correspondingly, scan-based processing is applied to the output of the static fusion process, enabling real-time surveillance results that exceed centralized (or single-sensor) processing results.

3.3 Track-extract-track

In May 2008, the N.R.V. Alliance participated in an experimental campaign in conjunction with the French company ACTIMAR, based in Brest, France. In particular, ACTIMAR acquired roughly 6hrs of HF radar data using its two shore-based radars. The HF radar dataset has proven to be a particularly challenging one, with large numbers of clutter-induced contacts.

Based on a first assessment, it appears that effective HF radar tracking is quite challenging. The radars yield few tracks in the more distant vessel sea lane: analysis of the vessel transponder (AIS) data revealed that only large tankers are tracked. There is greater success with the closer sea lane, though here again performance is limited. Many near-range vessels identified by AIS are not found in the HF radar tracks.

This dataset allows for an investigation of the advantages of the track-extract-track multi-stage tracking architecture. In this approach, we extract contact data from the first stage of tracking, and proceed with a second stage of tracking with the remaining contacts; the process can be iterated with a third stage of processing, and so on. The key advantage of the approach is that it allows for additional, weaker target tracks to be extracted from the data. This is accomplished by increasing the data correlation gates in the tracker, as well as lowering the track-confirmation criterion. In principle, a similar result could be achieved with centralized processing, but with a much more complex adaptive-tracking methodology.

Further details may be found in [8]. An illustration of track-extract-track processing results is given in figure 5. The different track colors correspond to different iterations in the track-extract-track processing; transponder data is shown in red.

A second application of the track-extract-track methodology will be discussed in the context of the MSTWG TNO datasets.

4 The MSTWG TNO Datasets

In previous work, we have documented NURC results with the MSTWG datasets provided by NURC, ARL:UT, and TNO [16-17]. Here, we first discuss results for the new dataset provided by TNO [18]: the TNO blind dataset. As the name suggests, the novelty of this dataset lies in the fact that ground truth information was not made available to the MSTWG until after a first round of analysis and an exchange of tracking results. The NURC results illustrated here have not been modified further with knowledge of ground truth.

The dataset includes three source-receiver configurations, with FM, CW, and both FM and CW contacts, respectively. As identified in our analysis, the dataset contains two targets. DMHT results were generated with the following nominal parameter settings:

- Contacts/scan: 10; track initiation: 2-of-2; max missed detections: 5; target maneuverability: 0.01m/s³; data association gate: 99%; n-scan: 0.

Unfortunately, there proved to be some errors in the generation of Doppler measurements in the CW contact files; thus, only positional information is utilized. Figures 6-8 illustrate the DMHT tracks obtained. Platform information is in red (source) and blue (receiver). Contact data is in black, and confirmed tracks are in magenta.

In an attempt to improve track continuity, we experimented with larger numbers of contacts files as well as hypothesis tree depths (n-scan) greater than zero. For example, figures 9-10 illustrate platform 3 results with 20 contacts per file, with an n-scan setting of zero and two, respectively. We note some modest improvement with multi-hypothesis processing.

More significantly, these results suggest that scan-based tracking, even with a fairly sophisticated fusion engine like the DMHT, still suffers track fragmentation that an operator can identify visually. Unfortunately, significant algorithmic tuning is required to reduce the fragmentation level. This fact has motivated an investigation into the applicability of the track-break-track multi-stage processing approach as a means to associating overlapping-in-time tracks from the same sensor. Indeed, we need a means to circumvent the fundamental assumption that a target is seen at most once in each scan of data: the track-break-track approach does so in a simple manner.

We apply the track-break-track methodology to platform 3 contact data and illustrate both single-stage and two-stage results in figure 11. We note that, while the results are similar, the fragmentation is reduced with the two-stage approach.
It is worth noting that this two-stage scheme, like all the two-stage schemes discussed in this paper, do not introduce any appreciable delay in fusion processing. Indeed, as previously discussed, all stages of DMHT processing are scan-based, and process data as it becomes available from the previous stage of processing.

A second potential application of two-stage processing in the context of the TNO datasets is relevant to the first MSTWG TNO dataset [19]. Figure 12 illustrates one of the tracking results reported in [17] related to this dataset. The key point to note it that the scenario includes a target.
that executes piecewise-linear trajectories, and that executes two maneuvers in very close proximity to fixed sources of clutter returns.

![Figure 12. DMHT tracks for the first MSTWG TNO dataset.](image)

Further, one of the clutter points has the same nominal SNR as the target. Thus, it is extremely challenging to disambiguate the target contacts from the fixed clutter contacts, as they are close in both kinematic and feature measurement spaces while the target is deviating markedly from nearly-constant velocity motion.

We conjecture that a customized version of the track-extract-track architecture, where the first DMHT module utilizes a nearly constant position model, may effectively disambiguate the returns. Our intuition is that in so doing we favor the solution with fixed clutter tracks, as maneuvering targets traverse the region. The solution so obtained should be compared with an IMM-MHT [1] in which the interaction between moving and fixed target modes is set close to zero; this too will bias the solutions away from track swaps, avoiding maneuvering targets being drawn to fixed clutter, and fixed clutter tracks being drawn to moving targets.

5 Tracker Performance Evaluation

A significant challenge in the target tracking community is the determination of a set of performance metrics that provide a sufficient characterization of algorithmic effectiveness. For instance, the performance of a detection system is fully characterized by the receiver operating characteristics (ROC) curve. Unfortunately, performance characterization at the output of a fusion and tracking system is more complex. In part, this is due to the more complex nature of the tracker output that includes an additional (time) dimension. This introduces the need for additional performance metrics like track fragmentation and, more significantly, it significantly complicates the mapping of tracks to targets. Further, some trackers are post-surveillance tools that work with significant time latencies, while others operate in real-time mode: this makes for difficult performance comparisons. In the end, there is no simple set of exhaustive measures of performance (MOPS) that are consistently accepted in the tracking community.

The Multistatic Tracking Working Group (MSTWG) was initiated in 2005 as a multi-laboratory collaboration on tracking and fusion algorithms. The group operates under the auspices of the International Society of Information Fusion (ISIF). Near the outset of the group, a widely accepted group of performance metrics was agreed upon [20]; these include track hold, false track rate, track fragmentation, and track localization error. These metrics have been applied to a number of tracking algorithms and common datasets, as reported on in special sessions at several past international conferences culminating with FUSION 2009. While the metrics provide an effective means for cross-algorithm comparisons, the metrics are hampered by a number of tracking phenomena that include time-over-death, track overlap, and track swap.

Tracker evaluation as prescribed in [20] requires that, subject to a global (average) localization threshold, tracks be partitioned as target-originated or not. The former set then provides the basis for the computation of track hold or track PD (track duration as fraction of target duration), fragmentation rate (ratio of number of true tracks to targets per unit time), and localization error (average discrepancy in meters between track positional estimate and target location). The latter set of tracks provides the basis for the false track rate or FTR (average number of false tracks per unit time).

An illustration of a tracker evaluation example is in figure 13. The evaluation includes several modifications to the metrics defined in [20]. First, we define fragmentation rate based solely on those targets that are successfully tracked. Thus, the metric translates to the notion of additional tracks per unit time. Second, the evaluation addresses the tracking phenomena listed above. A flowchart for performance assessment is in figure 14.

The example in figure 13 and the flow diagram in figure 14 identify additional performance metrics. Specifically, we have introduced an iterative scheme that breaks tracks when these are not consistently mapped to the same target. When a sufficient number of consecutive track updates is associated with a different target (or with none) relative to the global mapping of most frequent mappings, a track break is introduced. The methodology relies on a distance mapping threshold and a maximum-anomaly threshold. Additionally, if two tracks share the same target for more scans than the same maximum-anomaly threshold, a redundant track is identified. Thus, additional metrics are the broken track rate or BTR (number of breaks per unit time), as well as the redundant track rate or RTR (number of redundant tracks per unit time).

The computation of PD reflects track overlaps in that the determination of the time duration of true tracks avoids double counting. Further, the localization error
computation reflects track duration in that longer-duration tracks are weighed more heavily in the overall track error. Localization error is not defined in cases of track before target birth or after target death. Performance evaluation examples are in figures 15-16.

The tracker metrics as defined here provide an effective means to performance evaluation that addresses the difficulties associated with track-purity issues. Of course, the BTR metric is crucial, as it reflects the extent of tracker output manipulation prior to performance evaluation. An illustration of automated track manipulation is given in figure 17.

Figure 13. An illustration of tracker performance metrics for a specific tracking example that includes true tracks (blue) and false tracks (red).

Figure 14. Flowchart for robust performance evaluation.

Figure 15. Track break (right) in response to a significant track-truth mapping anomaly.

Figure 16. Track breaks for a significant anomaly (maximum anomaly threshold is 5); the embedded anomaly with target 3 is too short for a track breaks.

Figure 17. Visual illustration of automated track-breakage with initial track (left) and decomposition into a number of almost-pure track segments (right).

6 Conclusions

The NURC Distributed Multi-Hypothesis Tracker (DMHT) is a high-performance, computationally efficient, and modular algorithm that was initially developed for undersea surveillance. In past work, we have demonstrated that under certain conditions multi-stage processing outperforms centralized, single-stage processing (track-while-fuse): track-before-fuse in multi-rate, multi-sensor settings as well as in undersea surveillance with high detection thresholds and fading target effects; fuse-before-track in large sensor networks; track-extract-track in high-clutter HF radar tracking; and track-break-track in difficult crossing-target scenarios.
These results are seemingly at odds with fundamental results in the nonlinear filtering and distributed detection literature, and are based on the fundamental sub-optimality of all current approaches to target tracking that must contend with data-association uncertainty.

This paper provides an overview to the NURC DMHT and its application in numerous multi-stage configurations. Additionally, we provide new single-stage and multi-stage processing results for the MSTWG TNO blind dataset, the latter obtained with the track-break-track approach. Further, we motivate the use of the track-extract-track approach for the first MSTWG TNO dataset. Finally, we describe recent work in robust performance evaluation.

7 Acknowledgements
The authors wish to thank Pascal de Theije (TNO) for providing the two datasets used in the analysis reported in section 4. In addition, we thank our student interns Maxence Dingboe and Owen O’Neill from the French Naval Academy in Brest, who contributed to our work on tracker performance metrics documented in section 5.

References