An ML-MHT Approach to Tracking Dim Targets in Large Sensor Networks

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 – Poor individual sensor performance as well as a large number of sensor scans per time interval are two challenges for multi-target tracking is large sensor networks. We introduce a two-stage processing scheme (ML-MHT) to address the former issue, and another to address the latter issue (MHT^2). We consider as well the combination of these two techniques* (ML-MHT^2). *Simulation results are encouraging. Future work will include application of these techniques to more challenging multi-sensor datasets characterized by extremely poor detection and localization performance.

Keywords: Maximum likelihood, multi-hypothesis tracking, multi-sensor multi-target tracking, data association.

1 Introduction

This paper addresses the multi-target tracking problem in large sensor networks. We assume that the sensors are synchronized, in the sense that the scan times are common to all sensors. Conventional multi-hypothesis tracking methods are problematic in this setting for two reasons. First, individual sensors in large networks generally have modest or poor detection and localization performance. Accordingly, we introduce a *maximum likelihood* (ML) approach to improve the statistical quality of contact data, prior to MHT processing: we call this the *ML-MHT*.

Second, a very large hypothesis tree depth (in terms of number of sensor scans) is required to achieve a moderate time depth, as needed to disambiguate association hypotheses in multi-target settings. Thus, we consider the concatenation of MHT modules whereby the first module provides zero-time-duration tracks that associate contacts across synchronous sensor scans, and the resulting short-tracks are processed in the second MHT module: we call this the *MHT^2*.

We consider as well the combination of these two techniques: we call this the *ML-MHT^2*. We test these architectures with simulated multi-sensor data, and the results are encouraging. Future work will include application of these techniques to benchmark datasets provided by METRON as part of collaborative international multi-laboratory research that is ongoing in the ISIF *Multi-Static Tracking Working Group* (MSTWG).

A possible, alternative multi-stage MHT processing paradigm for dim targets would involve a first tracking stage with extremely small process noise and stringent track maintenance criteria, leading to highly fragmented, short-duration tracks. In a second processing stage, with larger hypothesis tree depth, relaxed track maintenance criteria, and higher process noise, the short-duration tracks would purportedly be associated into longer-duration maneuvering tracks.

This paper is organized as follows. Section 2 describes our maximum-likelihood preprocessing stage followed by MHT processing (the ML-MHT), and section 3 describes the concatenation of MHT processes (the MHT^2 and ML-MHT^2). Simulation results are provided in section 4. Conclusions and plans for future work are in section 5.

2 The ML-MHT

In past work, we introduced a *fuse-before-track* paradigm that includes static fusion via a grid-based approach, followed by MHT processing [1]. The grid-based approach has many limitations, most notably an inability to handle closely-spaced targets and poor performance for targets that are near the edge of a grid cell. Earlier investigations into more elegant approaches to static fusion included a scan-box approach, a *probabilistic hypothesis density* (PHD) approach, and a bootstrap maximum-likelihood approach [2-3]. Unfortunately, these methods provide inconclusive or poor performance, due to computational requirements, inherent sub-optimality, and, again, an inability to handle closely-spaced targets effectively.

Despite limited results to date, we believe the motivation for pursuing a *fuse-before-track* approach to multi-target tracking in large sensor networks with poor individual sensor performance is sound. The two-stage approach seeks to collapse a large number poor sensor scans into a single, equivalent scan of higher quality. (Techniques to do so that are insensitive to scan ordering are particularly appealing.) Following data reduction, a conventional scan-based tracker (MHT or otherwise) may be employed.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **JUL 2010**	2. REPORT TYPE	3. DATES COVERED **00-00-2010 to 00-00-2010**

4. TITLE AND SUBTITLE **An ML-MHT Approach to Tracking Dim Targets in Large Sensor Networks**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **NATO Undersea Research Centre,Applied Research Department,Viale S. Bartolomeo 400,19126 La Spezia, Italy,**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Presented at the 13th International Conference on Information Fusion held in Edinburgh, UK on 26-29 July 2010. Sponsored in part by Office of Naval Research, Office of Naval Research Global, and U.S. Army Research Laboratory's Army Research Office (ARO). U.S. Government or Federal Rights License.

14. ABSTRACT
Poor individual sensor performance as well as a large number of sensor scans per time interval are two challenges for multi-target tracking is large sensor networks. We introduce a two-stage processing scheme (ML-MHT) to address the former issue, and another to address the latter issue (MHT2). We consider as well the combination of these two techniques (ML-MHT2). Simulation results are encouraging. Future work will include application of these techniques to more challenging multi-sensor datasets characterized by extremely poor detection and localization performance.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**7**	

In this paper, we develop an effective ML approach as a first stage in *fuse-before-track* processing. There are two key aspects to the approach developed here. First, we avoid computationally costly numerical optimization schemes by evaluating the likelihood function *only* at contact locations. Having identified the top-scoring contact, one could imagine the following cumbersome methodology: (1) remove contact data due to a single target (roughly, remove a number of contacts equal to number of scans times the target detection probability); (2) collapse the extracted contacts into an equivalent fused contacts; (3) iterate the ML equations on remaining contacts. The second key aspect of our approach is to replace steps (1-3) with a simpler scheme whereby the top *M* contacts from the first set of ML evaluations are extracted, and each contact is kept within its original data scan. That is, we do not collapse the scans into a single, equivalent scan. In addition to the simplicity of our approach, the procedure (1-3) is potentially problematic in the case of closely-spaced targets.

We now proceed with a description of our ML approach to contact-data reduction. Future work will include an extension to non-linear measurement models and a comparison to the scan-collapse approach identified by steps (1-3) above.

We consider the case of linear measurements of two-dimensional target positional perturbed by additive Gaussian noise, as given by (2.1).

$$Z = X + w\,,\ \ w \sim N(0,\Sigma). \qquad (2.1)$$

The measurement covariance matrix Σ is assumed to be constant over all target-induced contacts. In each scan of data, all targets in the surveillance region are detected with probability *p*, and false contacts are uniformly distributed in this region (of area *u* [m^2]), with the number of false contacts Poisson distributed with mean λ.

For a given sensing time epoch, assume that N is the number of synchronous sensors, and let $n_i, 1 \le i \le N$ be the number of contacts from each sensor. The contacts from the i^{th} sensor are denoted by $Z_{ij}, 1 \le j \le n_i$. The likelihood function for target location is given by (2.2).

$$\Lambda(X) = \prod_{i=1}^{N} \left\{ \frac{1-p}{u^{n_i}} \mu_\lambda(n_i) + \frac{p \cdot \mu_\lambda(n_i - 1)}{u^{n_i - 1} n_i} \sum_{j=1}^{n_i} p(Z_{ij} \mid X) \right\}, \qquad (2.2)$$

$$\mu_\lambda(n) = \frac{\lambda^n \exp(-\lambda)}{n!}\,, \qquad (2.3)$$

$$p(Z \mid X) = \frac{1}{2\pi |\Sigma|^{0.5}} \exp\left(-\frac{1}{2} (Z - X)' \Sigma^{-1} (Z - X) \right). \qquad (2.4)$$

We evaluate $\Lambda(\cdot)$ for $X \in \{Z_{ij}, 1 \le i \le N, 1 \le j \le n_i\}$. We identify the top M likelihood function evaluations; the contacts corresponding to these contacts are kept, and all others are discarded, leading to a *thinned* version of the *N* sets of contacts. In particular, we now have $M << \sum_i n_i$ contacts for a given time epoch. It is important that we select $M > p \cdot \lambda_T \cdot N$, where λ_T is the expected number of targets.

The ML approach defined here is consistent with processing paradigms that invoke *hard data association*, since at no stage here is there is a weighted merging of contact data [4].

The significantly smaller contact files that result from the ML processing scheme described here constitute the input to an MHT processing stage. Our approach to track-oriented MHT is based on [5-6].

3 The MHT^2 and ML-MHT^2

It is important for MHT processing extend over a reasonable time extent. This is quite problematic in large sensor fields, where there are many data scans in short-duration time intervals. This is true whether or not we proceed with the ML processing stage described in section 2.

Accordingly, it is imperative to reduce significantly the number of track hypotheses prior to MHT process with large tree depth (i.e. large *n-scan*). A straightforward methodology to enable this is described here. As a reminder, we assume that the sensors are synchronized so as to have the same sequence of scan times. (Our methodology could be extended to handle the more general case of non-synchronous sensors, though we do not consider this here.)

In a first MHT processing stage (with small or zero *n-scan*), we associate contacts into tracks, though with track termination for non-zero time increments. That is, we perform automatic tracking separately for each (synchronous) collection of sets of contacts, with no track continuity between these collections. The resulting tracks exist over multiple sensor scans, but have zero time duration. As a byproduct of this process, tentative tracks that fail to achieve the track confirmation threshold are discarded.

The second MHT processing stage contends with a vastly simpler data-association task that associates short-duration tracks over time. In particular, this second stage can easily handle large hypothesis tree depths (*n-scan*>>0) with modest computational expense. The large *n-scan* allows for non-zero time-depth reasoning in large sensor networks. This capability is particularly useful in dense target scenarios with non-trivial target disambiguation.

4 Simulation Studies

In this section, we provide illustrations and preliminary performance assessment of the multi-stage MHT architectures introduced in sections 2-3.

4.1 ML-MHT vs. MHT

We start with a scenario that includes a single maneuvering target. Scenario and tracker parameters are in tables 1-2. We have four variations on the MHT, and two on the ML-MHT. In particular, we have two different MHT track-initiation settings, and two *n-scan* values for all cases.

Table 1. Simulation settings in single-target performance study.

Simulation parameter	Setting
Monte Carlo iterations	20
Number of scans	30
Scan interval	5sec
Surveillance region	x_{min}=-100, x_{max}=100, y_{min}=-20, y_{max}=20
Target start location	x=-75, y=-5
Target velocity (1st half)	v_x=1, v_y=0.1
Target velocity (2nd half)	v_x =1, v_y =-0.1
Number of sensors	40
FAR	10
PD	0.5
Contact localization error	5m (both x and y)
Track localization threshold	10m

Table 2. Key MHT and ML-MHT settings.

Tracker	M-of-N	n-scan
MHT-1	22-of-40	0
MHT-2	22-of-40	2
MHT-3	25-of-40	0
MHT-4	25-of-40	2
ML-MHT-1	10-of-40	0
ML-MHT-2	10-of-40	2

In all cases, we have a target process noise of $0.001m^2s^{-3}$, termination after 40 missed detections, a target prior velocity standard deviation of $1ms^{-1}$ in both dimensions, and a 99% data association gate. For ML processing, we have M=20 (see section 2).

Performance results are given in table 3. The results suggest comparable detection (PD, FAR) and localization performance (LE – localization error) for the two architectures, perhaps slightly better with the ML-MHT. Interestingly, the ML-MHT exhibits higher FR (fragmentation rate per hour), though absolute fragmentation is less than 3. The ML-MHT is most interesting from a timing rate, or TR perspective (i.e. execution time divided by scenario time).

Table 3. Monte Carlo performance results in single-target performance study.

Architecture	PD	FAR	FR	LE	TR
MHT-1	0.71	120.0	1.0	5.84	0.67
MHT-2	0.56	133.2	1.0	5.94	2.60
MHT-3	0.72	13.20	1.0	3.55	0.65
MHT-4	0.73	26.40	1.0	4.14	2.60
ML-MHT-1	0.86	91.20	44.0	3.49	0.23
ML-MHT-2	0.87	88.80	43.0	3.46	0.25

As tracker performance depends on parameter tuning and the results do not favor either architecture convincingly, we move next to a setting where ML-MHT may be of greater interest.

4.2 Very Large Networks: A Case for the ML-MHT

Even for *n-scan*=0, scan-based processing is problematic in *very* large sensor networks, due to the need for TR<1 for real-time processing. We examine here the case of 100 sensors; all other simulation parameters are as in table 1. Tracker parameters of note are in table 4. In both architectures, we have termination after 100 missed detections. For ML processing, we have M=50 (see section 2).

Table 4. Key MHT and ML-MHT settings.

Tracker	M-of-N	n-scan
MHT (red)	50-of-100	0
ML-MHT (cyan)	20-of-100	0

We now have PD~0.95, FAR=0, and LE~1.83m for both architectures; also, we have FR=1 (MHT) and FR=25 (ML-MHT), i.e. absolute fragmentation is 2 in the latter case. Thus, we find that both architectures perform much better than previously, thanks to the much larger sensor network. However, we now have TR=3.64 (MHT) and TR=0.84 (ML-MHT). Thus, while the ML-MHT still satisfies the real-time processing requirement, the MHT fails to do so. An illustration of one realization is given in figure 1.

It is interesting to examine performance in multi-target settings, where the need for hypothesis tree depth with non-zero time depth is critical. We do so next.

4.3 Closely-Spaced Targets: The MHT^2 and $ML\text{-}MHT^2$

The MHT and ML-MHT are severely challenged in multi-target scenarios in large sensor networks, due to their inability to utilize a large *n-scan*. We explore this setting with a modified scenario, with two non-maneuvering targets as indicated in table 5. Tracker architectures and key parameters are specified in table 6, and simulation results are in table 7.

Table 5. Multi-target modifications to settings in table 1.

Simulation parameter	Setting
Monte Carlo iterations	50
Number of targets	2
Target one start location	x=-75, y=-5
Target one velocity	v_x=1, v_y=0.1
Target two start location	x=-75, y=5
Target two velocity	v_x =1, v_y =-0.1
Contact localization error	1m (both x and y)
Track localization threshold	2m

Table 6. Tracker settings.

Tracker	M-of-N	n-scan
MHT (red)	20-of-40	0
ML-MHT (blue)	10-of-40	0
ML-MHT^2 (yellow)	10-of-40	0 (1st stage), 200 (2nd stage)
MHT^2 (green)	20-of-40	0 (1st stage), 200 (2nd stage)

No track confirmation stage is applied to second-stage MHT processing in both the ML-MHT^2 and MHT^2, i.e. no first-stage tracks are discarded. For all architectures, termination is after 40 missed detections (for ML-MHT^2 and MHT^2 this applies to the *second* stage, while as noted previously first-stage termination is at non-zero time increments). For ML processing, we have M=40 (see section 2).

Table 7. Simulation results.

Architecture	PD	FAR	FR	LE	TR
MHT (red)	0.24	154.08	1.00	0.57	0.68
ML-MHT (blue)	0.64	34.09	1.65	0.55	0.11
ML-MHT^2 (yellow)	0.58	24.00	8.02	0.62	0.18
MHT^2 (green)	0.29	62.40	14.44	0.58	0.43

All architectures perform similarly in terms of track localization. Preliminary indications are that the MHT architecture is weakest in terms of track detection, while the ML-MHT^2 is strongest. Indeed, we find that the ML-MHT suffers as a result of track swap (leading to false track classification), while the MHT^2 suffers as a result of high input clutter that is not reduced through ML processing, leading to track loss and limited track hold. The ML-MHT^2 ameliorates both effects.

An illustration of the need for large downstream *n-scan* is exemplified in figure 2. Here, we see that the ML-MHT^2 is able to track successfully, while the ML-MHT incurs a track swap. A further illustration of the need for large downstream *n-scan* and of the effectiveness of upfront ML processing is in figures 3-4, where we see a scenario realization with results for all four architectures under study. We see false track formation with the MHT, track swap with the ML-MHT, successful tracking with the MHT^2 (though with limited track hold), and successful tracking with the ML-MHT^2.

4.4 Robustness of the ML-Enhanced Architectures

The robustness of tracker performance with respect to parameter settings is important. In particular, the ML-MHT introduces a new parameter that reflects the expected number of targets in the surveillance region. Indeed, recall that we require $M > p \cdot \lambda_T \cdot N$, where p is the detection probability, λ_T is the expected number of targets, and N is the number of scans to which ML processing is applied. Thus, it is important that the choice of M be sufficiently large. What is the impact of too large a choice?

In figures 7-8, we illustrate results for a scenario realization that is identical to those in section 4.3, with the sole modification that we have set M=60, i.e. consistent with the presence of three targets rather than two. There is no notable change in tracking results. Figures 5-6 illustrates one such realization, with MHT and ML-MHT tracks displayed with the usual color conventions (red, blue respectively).

5 Conclusion and Further Extensions

Poor individual sensor performance as well as a large number of sensor scans per time interval are two challenges for multi-target tracking is large sensor networks. In this paper, we introduced a two-stage processing scheme (ML-MHT) to address the former issue, and another to address the latter issue (MHT^2). We considered as well the combination of these two techniques *(*ML-MHT^2*)*. Simulation results are encouraging, and suggest that ML processing is useful to reduce false track formation, while repeated MHT processing (allowing for large downstream *n-scan*) allows for successful multi-target disambiguation.

Future work will include application of these techniques to more challenging multi-sensor datasets characterized by extremely poor detection and localization performance. In particular, we will examine multi-stage processing of simulated multistatic sonar data made available to the ISIF *Multi-Static Tracking Working Group* (MSTWG) by METRON [7]. This work will require nonlinear extensions to the ML processing stage, and relies on precise transformation of bistatic sonar measurement errors to Cartesian coordinates [8-9]. Illustrations of preliminary results on the METRON datasets (obtained with the MHT processing architecture) are in Annex A. These results give some appreciation for the challenge associated with effective track extraction in low-quality data.

An additional area for future work includes a comparison of our ML processing with a modified approach that would involve greedy contact extraction and fused contact determination (steps 1-3 in section 2).

Finally, it is of interest to extend the architectures introduced here to handle non-synchronous sensor networks, where there is no requirement that scan times be matched across sensors.

References

[1] S. Coraluppi, M. Guerriero, C. Carthel, and P. Willett, Fuse-before-Track in Large Sensor Networks, to appear in *ISIF Journal of Advances in Information Fusion*.

[2] M. Guerriero, S. Coraluppi, and P. Willett, Analysis of Scan and Batch Processing Approaches to Static Fusion in Sensor Networks, in *Proceedings of the SPIE Conference on Signal and Data Processing of Small Targets*, Orlando FL, USA, March 2008.

[3] R. Bethel and K. Bell, “A multi-hypothesis glrt approach to the combined source detection and direction of arrival estimation problem,” *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP ’01),* Salt Lake City, UT, USA, May 2001.

[4] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, 1999.

[5] S. Coraluppi, C. Carthel, M. Luettgen, and S. Lynch, All-Source Track and Identity Fusion, in *Proceedings of the National Symposium on Sensor and Data Fusion*, San Antonio TX, USA, June 2000.

[6] S. Coraluppi and C. Carthel, Distributed Tracking in Multistatic Sonar, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41(3), July 2005.

[7] K. Orlov, Description of the ‘MetronSimulation’ data set for MSTWG, *METRON Technical Memorandum*, June 2009.

[8] S. Coraluppi, Multistatic Sonar Localization, *IEEE Journal of Oceanic Engineering*, vol. 31(4), October 2006.

[9] S. Coraluppi, C. Carthel, D. Hughes, A. Baldacci, and M. Micheli, Multi-Waveform Active Sonar Tracking, in *Proceedings of the Third International Waveform Diversity & Design Conference*, Pisa, Italy, June 2007.

A The MSTWG METRON Datasets

The MSTWG METRON data includes five scenarios. The data release included target ground truth for the first scenario. Thus, we limit our discussion to this scenario. In particular, the scenario includes four targets that execute repeated square-like motion patterns. There are 25 source-receiver combinations at each scan time, with active sonar scans of either FM or CW (bistatic Doppler-enhanced measurements). Details of the sensor configuration and data characteristics are in [7]. Here, we illustrate preliminary processing results for this scenario.

A fair amount of MHT parameter tuning was required in order to extract reasonable tracks when compared with the known ground truth target trajectories. Table 8 summarizes the key MHT parameter settings. Figures 7-11 illustrate our results. (Note that numerous target square-like motion patterns are performed, so multiple tracks for each target are seen in a consolidated view.)

Table 8. Optimized MHT settings for the MSTWG METRON datasets.

Parameter	**Setting**
FM SNR threshold	9dB
CW SNR threshold	5dB
Range threshold	22km
TDOA threshold	40sec
Track initiation	3-of-175
Track termination	15min
Hypothesis tree depth	200
Target velocity prior standard deviation (in both *x* and *y*)	$6ms^{-1}$
Target process noise	$0.0075m^2s^{-3}$

It is worth noting that, given the extremely large cross-range localization errors at significant ranges from the receivers, we have introduced additional tracker functionality that discards all contacts beyond a specified range to the receiver, or, alternatively, beyond a specified *time difference of arrival* (TDOA). (Note that these thresholds are related, but do not have exactly the same effect in the bistatic case.)

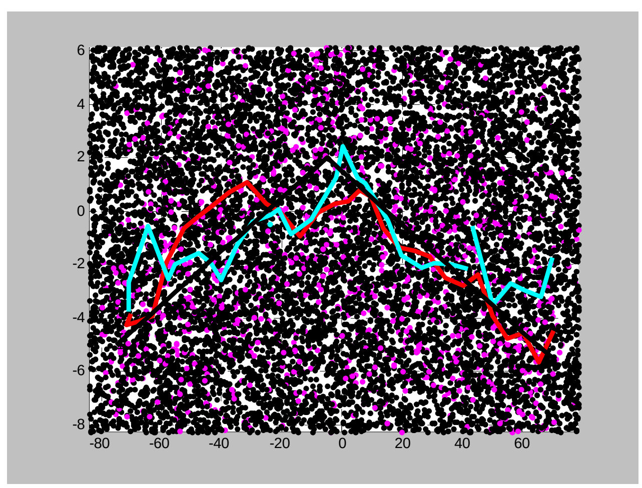

Figure 3. One realization of contact data (magenta, black dots, where magenta is target-originated) based on 100 sensors. Black line is target trajectory. MHT (red) and ML-MHT (cyan) tracks are shown as well.

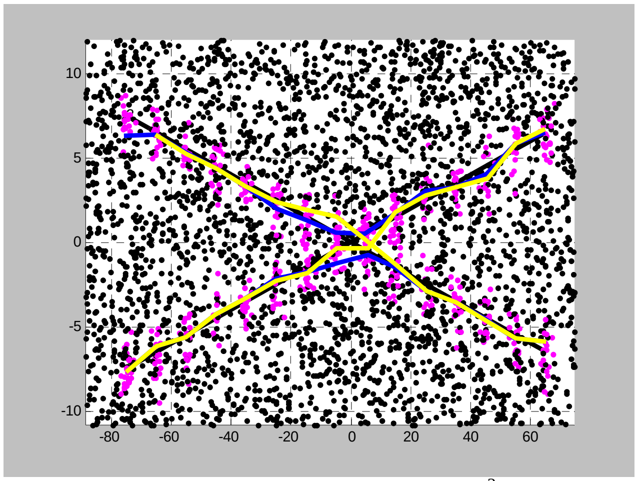

Figure 2. ML-MHT (blue) and ML-MHT^2 (yellow), illustrating the success of the latter to avoid track swap in dense target scenarios.

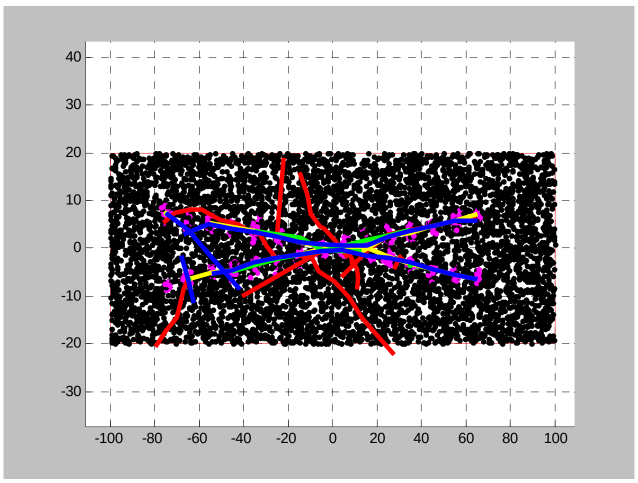

Figure 3. A realization of MHT (red, ML-MHT (blue), ML-MHT^2 (yellow), and MHT^2 (green), illustrating the success of the latter two in handling track swap, and of the ML-based architectures to suppress the damaging effects of clutter.

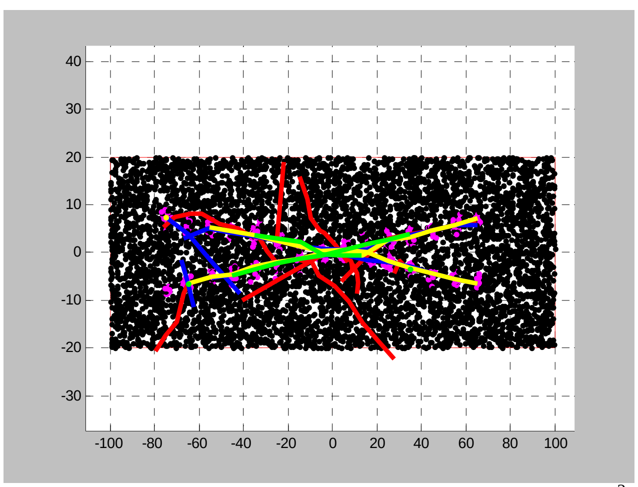

Figure 4. Same realization as in figure 5, with ML-MHT^2 (yellow) more visible.

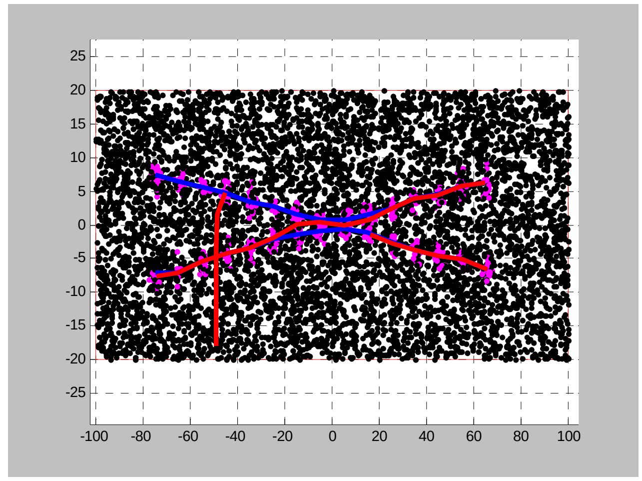

Figure 5. An MHT (red) and ML-MHT (blue) realization with poor settings in ML-MHT, illustrating its robustness.

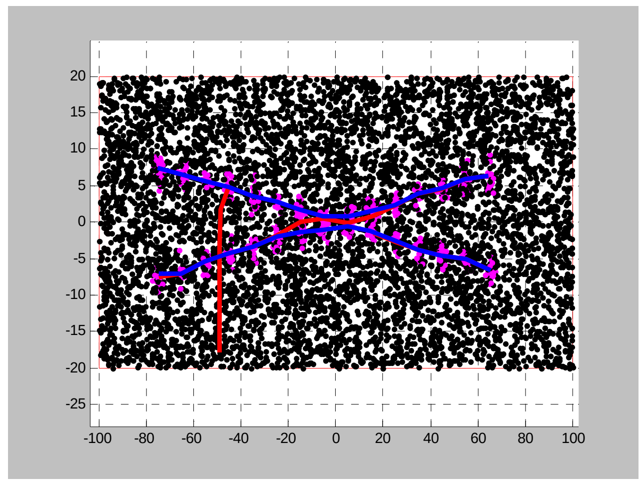

Figure 6. Same realization as in figure 7, with the ML-MHT (blue) more visible.

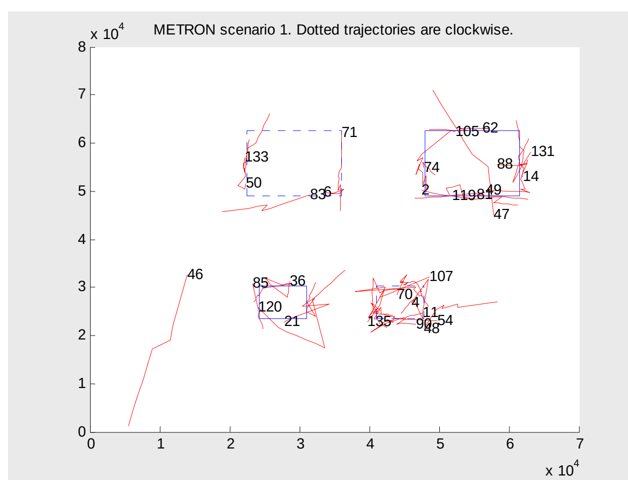

Figure 7. Optimized MHT output (red) for the first MSTWG METRON dataset, for which ground truth is known (blue).

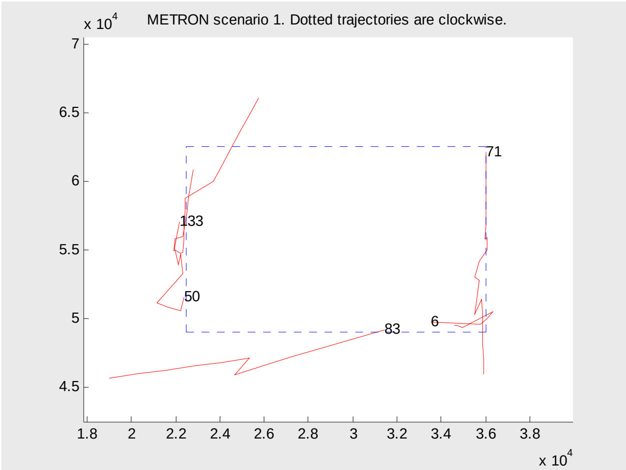

Figure 8. Same result (close-up view of first target).

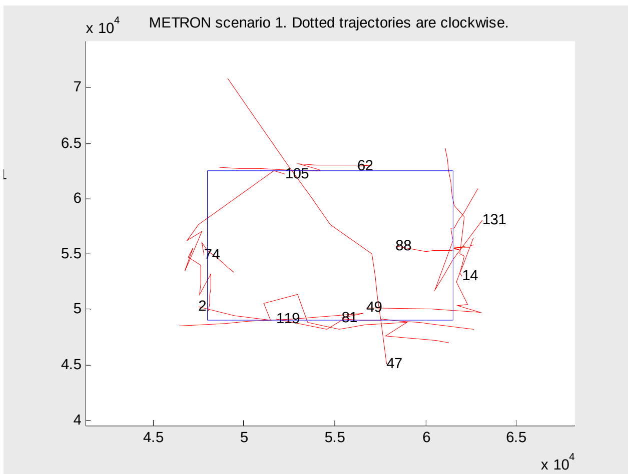

Figure 9. Same result (close-up view of second target).

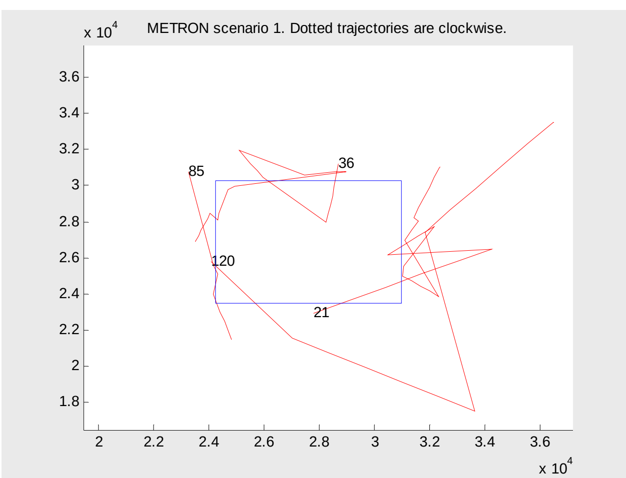

Figure 10. Same result (close-up view of third target).

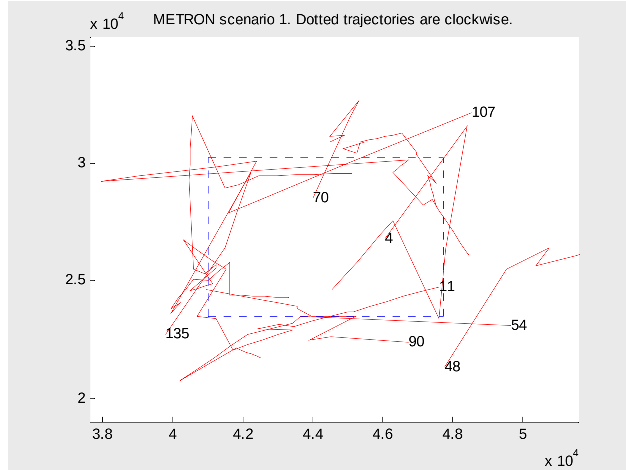

Figure 11. Same result (close-up view of fourth target).