Multi-Hypothesis Sonar Tracking

Stefano Coraluppi
Anti-Submarine Warfare Department
NATO Undersea Research Centre
Viale S. Bartolomeo 400, 19138 La Spezia, Italy
coraluppi@saclantc.nato.int

Craig Carthel
Command and Operational Support Department
NATO Undersea Research Centre
Viale S. Bartolomeo 400, 19138 La Spezia, Italy
carthel@saclantc.nato.int

Abstract - *This paper introduces a multi-hypothesis multistatic sonar tracker for undersea surveillance. Multistatic sonar increases the data rate and has the potential to improve surveillance capabilities, provided effective target tracking is performed. Our multi-hypothesis tracker includes features not generally found in other multi-hypothesis trackers. Data association is based on an efficient linear programming approach, to which we introduce a novel modification that improves track continuation. We use equality constraints in the LP, and tracks are removed when they fail a confirmation criterion. Short duration tracks are classified as false and removed. System and measurement uncertainties are reflected through multistatic contact covariances. This uncertainty impacts the data association hypotheses that are considered, as well as their log-likelihood scores. We test the improved performance of this tracker over our earlier baseline tracker, with a number of benchmark examples of interest and through Monte Carlo evaluation.*

Keywords: Multi-sensor multi-target tracking, active sonar, multistatic sonar, undersea surveillance.

1 Introduction

This paper studies a multi-hypothesis multistatic tracking algorithm. The multi-hypothesis approach to tracking utilizes a deferred approach to data association leading to improved performance.

Our previous work in multistatic tracking uses a baseline algorithm [1-2]. In addition to a detailed description of the baseline tracker, these references include extensive performance analysis with simulated and sea trial data.

In Section 2, we first provide a summary of the baseline tracking algorithm, and we then describe the multi-hypothesis tracker. This tracker includes a number of novel features that extend existing sensor fusion and target tracking research. These include an efficient scheme for the determination of near-optimal data assignments, a novel track scoring mechanism, and the use of multistatic contact data with precise accounting of measurement errors. An overview of past research in sensor fusion and target tracking can be found in [3-5].

As with our baseline multistatic tracker, the multi-hypothesis tracker is designed to process contact files produced by the sonar signal and information processing chain at NURC [6]. These files provide a compact representation of detection information and have been demonstrated to support real-time data exchange [7]. Furthermore, the data content provides an ideal input to multi-platform fusion and tracking algorithms.

In Section 3, we describe our performance metrics of interest. We compare the baseline and multi-hypothesis trackers for some representative benchmark examples as well as with randomly generated scenarios, with simulated contact data. Our testing establishes the improved statistical performance of the multi-hypothesis tracker.

It should be noted that the contact data accounts for numerous sources of measurement error including time, bearing, array heading, source and receiver locations, and speed of sound. The supporting measurement covariance matrix expressions are derived in [8].

2 Target Tracking Algorithms

In this section, we first provide a summary of the baseline tracking algorithm, and then describe the multi-hypothesis tracker. This tracker includes a number of novel features that extend existing sensor fusion and target tracking research.

Our input to the tracking algorithm is a sequence of time-ordered contact files; each contact file corresponds to a specific source and receiver pair. In the multistatic case, where several sources may ping at the same time, this sequence of files may include a number of files with the same time stamp. Both the baseline tracker and the multi-hypothesis tracker arbitrarily order contact files with the same time stamp, and files are processed sequentially.

The main goal of an automated tracking algorithm is to drastically reduce the amount of data that an operator must contend with, by removing false contacts and associating target contacts to form tracks. Spurious false contacts often occur randomly in location, and thus will generally not lead to track formation. Contacts due to fixed clutter features and to targets occur with consistency, and generally lead to track formation. In addition to providing a performance improvement in a ROC curve sense, target tracking can significantly reduce the localization accuracy that is present in the raw contact data, by smoothing detections over time using target kinematic and measurement models.

2.1 Baseline Tracker

Each contact in the first contact file initiates an active track. Subsequently contact files are processed as follows. At each step, all active tracks are ordered based on the number of contacts on which they are based: active tracks are then processed in sequence. For each track, an association criterion is applied to the data whereby only sufficiently close contacts are considered, based on a gating threshold as well as the uncertainty in the predicted track range and bearing from the receiver, and the range and bearing contact uncertainty. If at least one contact passes the gating threshold, the statistical nearest neighbor is used to update the track. The contact is then discarded, and is unavailable to other tracks. After all active tracks are processed, the remaining contacts initiate new tracks.

There are two mechanisms by which active tracks are terminated. First, if a track fails to associate M contacts out of the first N scans of data, the track is terminated. Once a track associates M contacts within N scans of data, it is confirmed. Only confirmed tracks are provided to the tracker output. Once a track is confirmed, it is subsequently terminated when it fails to update for K consecutive scans of data.

Track location and velocity estimates are updated based on an extended Kalman filter (EKF). Both the association criterion and the EKF utilize a 2D measurement vector that includes range and bearing from the receiver. The measurement covariance matrix accounts for numerous sources of measurement error including time, bearing, array heading, source and receiver locations, and speed of sound [8].

In addition to kinematic measurement information, sonar contacts include feature information that includes SNR, time extent, beam extent, etc. This feature information is not currently exploited in our tracking algorithms. When processing the contact files, we do apply an SNR threshold to the data to remove low-SNR contacts.

Our kinematic filtering is based on a single nearly constant velocity motion model. As a result, the baseline tracker will track moving targets as well as fixed clutter points. At present, we do not classify tracks based on their velocity characteristics. On the other hand, we have implemented a simple track classifier whereby confirmed tracks that do not achieve a minimum length L, defined as the number of scans for which the track is confirmed, are classified as non-targets.

2.2 Multi-Hypothesis Tracker

The multi-hypothesis tracker uses the same kinematic and measurement models, association gate criterion, nonlinear kinematic filter, and track-length classifier as the baseline tracker. For further details on these aspects of the algorithm, see [1-2]. As noted above, enhanced measurement covariance matrix expressions that account for numerous sources of measurement error including time, bearing, array heading, source and receiver locations, and speed of sound are derived in [8]. Here, we will focus on the novel aspects of our multi-hypothesis tracker: the enhanced track formation logic, the enhanced data association scheme for track continuation, and the slightly revised track termination logic.

Unlike the baseline tracker, the multi-hypothesis tracker uses a true sliding window *M-of-N* track confirmation scheme. Rather, the baseline tracker terminates an unconfirmed track that fails to achieve the *M-of-N* criterion. The following example clarifies the difference in these approaches.

Track Confirmation Example. Consider a 3-*of*-4 confirmation criterion, and the following sequence of associated detections (denoted 'x') and missed detections (denoted 'o'): (x, o, x, o, x, x, o, x, …). The multi-hypothesis tracker will confirm the track at the 6th scan, while the baseline tracker will confirm the track at the 8th scan.

In the multi-hypothesis tracker, we introduce the notion of a track *score*. This is a log-likelihood score that is updated recursively as follows. The track score initiation is given by:

$$l(1) = s_0, \tag{1}$$

where the term in equation (1) is typically set to be a large negative quantity, which penalizes the initiation of new tracks. Subsequently, track score updates are given by equation (2) when a contact is used, and by equation (3) otherwise.

$$l(k+1) = l(k) + \log\left[p(t_{k+1} - t_k)\, g\left(X(k+1), P(k+1|k), Y_{k+1}\right)\right], \tag{2}$$

$$l(k+1) = l(k) + \log\left[p(t_{k+1} - t_k)\right]. \tag{3}$$

In equations (2-3), the term $p(t_{k+1} - t_k)$ is a track aging term that reflects the probability of continued target life, based on an underlying continuous-time Markov chain model with *target* and *no-target* states, and transition rate λ. In particular, the term is given by:

$$p(t_{k+1} - t_k) = \exp\left(-\lambda(t_{k+1} - t_k)\right). \tag{4}$$

In equation (2), the term $g\left(X(k+1|k), P(k+1|k), Y_{k+1}\right)$ reflects the likelihood that the measurement Y_{k+1} is associated with the track with predicted state estimate $X(k+1|k)$ and predicted state covariance $P(k+1|k)$. In particular, we have:

$$g\left(X(k+1|k), P(k+1|k), Y_{k+1}\right) = \frac{1}{(2\pi)^{3/2} \det\left(CP(k+1|k)C' + R\right)^{1/2}} \cdot \exp\left(-\frac{1}{2}\left(Y_{k+1} - Y(k+1|k)\right)\left(CP(k+1|k)C' + R\right)^{-1}\left(Y_{k+1} - Y(k+1|k)\right)\right) \tag{5}$$

where C is the linearized observation matrix evaluated at $X(k+1|k)$, R is the measurement covariance associated with measurement Y_{k+1}, and $Y(k+1|k)$ is the expected measurement based on the state prediction $X(k+1|k)$. Further details on the extended Kalman filter (EKF) used here can be found in [1-2].

Rather than making an immediate assignment of contacts to tracks, in the multi-hypothesis tracker a set of active tracks and a set of contacts lead to a set of track *hypotheses*. In particular, each contact leads to a new track hypothesis, and each active track leads to a set of track hypotheses, one corresponding to no contact, and one for each of the contacts that satisfies the association gate criterion. More generally, a set of track hypotheses and a set of contacts lead to a new set of track hypotheses, based on the same hypothesis generation methodology. The approach described here is a track-oriented MHT scheme, as opposed to the hypothesis-oriented MHT scheme; the track-oriented approach has significant computational advantages, as it does not require the explicit enumeration of global association hypotheses [3].

A key parameter in multi-hypothesis tracking is *N-scan*, i.e. the number of layers in the set of track hypothesis trees. With a latency specified by *N-scan*, all association hypotheses are *resolved*, i.e. a single global hypothesis is selected. This selection is determined as follows. Each track hypothesis has a track score, as defined above. The selection of a global hypothesis amounts to the selection of a subset of all track hypotheses, which accounts for all contacts exactly once. This problem can be cast as an integer-programming problem, for which an efficient linear programming relaxation approach leads to a near-optimal selection of a global hypothesis [9-10]. Having selected a global hypothesis, all track hypotheses that are inconsistent with this global hypothesis are removed.

Figure 1 summarizes the framework for the multi-dimensional assignment problem. Each row in the matrix equation corresponds to one branch in the set of track hypothesis trees. The vector c consists of the track likelihood scores; the matrix A is composed of ones and zeros (a one for component mn indicates that contact n is in track hypothesis m). The vector b is a vector of ones; this insures that each contact is accounted for exactly once in each feasible solution x, i.e. in each global hypothesis. After solving the linear program, the solution $\tilde{x}$ is used in place of the track scores, and a greedy assignment scheme is then used, whereby high-score tracks are successively chosen and competing tracks are removed. This results in a feasible solution to the integer program.

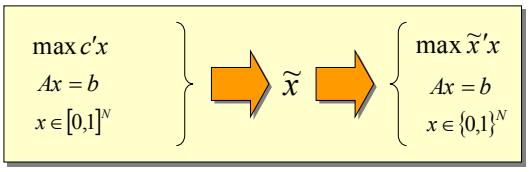

Fig. 1. Framework for efficient data association.

In addition to the track scores discussed above, which are relevant to the solution of the data association problem, a secondary track score is defined which is initialised at zero at the time of the last contact update, and propagates according to equation (3). This score is compared against a termination threshold, and tracks are terminated if their score falls below the threshold. This ensures that tracks that fail to update for a sufficiently long time are terminated. A track hypothesis that is a terminated track cannot subsequently gate with any contacts, and remains terminated. When a terminated track hypothesis reaches the *N-scan* layer, i.e. when a termination decision is resolved, the corresponding active track is terminated.

The track scoring and data association scheme discussed above generally lead to an assignment that minimizes the number of track objects consistent with the data. However, an undesirable aspect of the problem formulation is that no priority is given to the extension of confirmed tracks over the extension of active, but unconfirmed tracks. A simple change to equation (3) for track scoring purposes (but not for the secondary track score), is to introduce a pseudo track detection term P_{DT}, which penalizes the lack of a confirmed track update as follows:

$$l(k+1) = l(k) + \log\left[\left(1 - P_{DT}\right)p(t_{k+1} - t_k)\right]. \qquad (6)$$

We set P_{DT} close to one, which amounts to solving the global assignment problem for confirmed tracks only, and then fixing the assignment and solving a second global assignment problem with unconfirmed tracks and the remaining contacts.

As will be shown, the multi-hypothesis approach to data association leads to improved tracking performance. In fact, even with *N-scan* set to zero, the near-optimal assignment based on log-likelihood track scoring provides better data association (in a statistical sense) than is obtained in the greedy assignment scheme used in the baseline tracker, where track-to-contact assignments are made sequentially rather than with a global assignment scheme.

As with the baseline tracker, we use a simple track classifier whereby confirmed tracks that do not achieve a minimum length L, defined as the number of scans for which the track is confirmed, are classified as non-targets.

Our multi-hypothesis tracker includes a number of features not generally found in other multi-hypothesis trackers. First, our track scoring does not use log-likelihood *ratios*. Consistent with this fact, we do not discard contacts as part of our data association methodology. Rather, after data association, contacts are discarded when they are not part of a confirmed track, which successfully passes the *M-of-N* criterion. The novel modification with the pseudo track detection term P_{DT} improves the continuation of confirmed track.

3 Tracker Performance Evaluation

In this section, we first describe the performance metrics for the input data to the tracker as well as for the tracker output. Then, we compare the performance of the baseline tracker and the multi-hypothesis tracker with simulated data, with both deterministic target trajectories that explore specific benchmark cases of interest, as well as with stochastically generated target motion.

3.1 Measures of Performance

The quality of the input data to the tracker is defined in terms of detection performance and localization accuracy. Specifically, we define the following quantities:

- *Probability of detection* (PD): The ratio of the number of target detections and the number of detection opportunities.
- *False alarm rate* (FAR): The ratio of the number of non-target detections and the scenario duration (in minutes). The scenario duration is defined as the difference in time stamps of the first and last contact files.
- *Localization error* (RMSE): The RMS error between target detections and the ground truth acoustic reconstruction at that time.

An evaluation of the quality of the tracker output requires that we classify tracks as true or false. A track is defined as *true* if it associates more target detections than non-target detections. Otherwise, the track is defined as *false*. Having classified tracks in this manner, we then define the following quantities:

- *Track probability of detection* (T-PD): The ratio of the total time duration of true tracks and the scenario duration. For each true track, the time duration is defined as the difference in time of the first and last contact that the track associates; note that this applies only to the confirmed portion of the track that is seen at the tracker output, i.e. the track initiation phase before the M-of-N criterion is satisfied is not included.
- *Track false alarm rate* (T-FAR): The ratio of the number of false tracks and the scenario duration (in hours).
- *Track localization error* (T-RMSE): The RMS error between track location estimates and the ground truth acoustic reconstruction at that time.

There are numerous metrics that one can define and evaluate to quantify tracking performance. A key motivation for using the metrics defined above is that they can be easily related to the input metrics, along for a direct comparison and an understanding of the value-added of target tracking.

It should be noted that the FAR and T-FAR metrics are defined *per minute*, rather than *per scan* (or contact file). Indeed, as we are interested in comparing different choices of input data that correspond to different data rates, it is important to have operationally relevant metrics.

As defined, the T-PD is arguably a slightly pessimistic metric. Indeed, if the tracker were to operate as a simple pass-through and simply declare all contacts as tracks, T-FAR would equal FAR and T-RMSE would equal RMSE, while T-PD would be zero. This follows from the fact that point tracks have zero time duration, and thus do not contribute to the evaluation of T-PD. Thus, it could be argued that our T-PD metric slightly underestimates the effective time duration of tracks. On the other hand, it is possible that two tracks corresponding to the same target overlap slightly in time, and both tracks will contribute to the evaluation of T-PD.

An additional tracking metric that is often of interest is track fragmentation, which is defined below. This metric is indirectly reflected in T-PD. Indeed, a high fragmentation rate leads to many periods of time during which tracks are dropped, and subsequent tracks are not yet confirmed. Thus, high fragmentation generally leads to a decreased T-PD.

- *Track fragmention* (T-FRAG): The ratio of the number of tracks to the number of targets.

3.2 Data Simulation

We use a MATLAB-based data simulation capability, which allows for an arbitrary number of targets and source/receiver platforms in an *x*-*y* coordinate system, and an arbitrary scenario time duration. Each source/receiver platform is given an initial position and constant velocity. Targets trajectories are set deterministically or stochastically. In the first case, each target is given an initial position and constant velocity; note that this allows for the inclusion of fixed clutter points with zero velocity. Alternatively, we can allow initial target locations to be randomly selected within the region of interest, and the initial velocity is chosen according to a prior velocity distribution. Subsequently, target motion evolves according to a nearly constant velocity motion model, with fixed process noise parameters.

For each source/receiver platform, we specify a ping schedule, given by an initial ping time and a ping repetition time. Each ping leads to a number of contact files equal to the number of receivers. For each contact file, each target detection is governed by a fixed detection probability parameter, as well as the feasibility of target observation given the source-target-receiver geometry and the ping repetition time. (For example, if the ping repetition time is 1 minute, only targets for which the source-target-receiver signal travel time is less than 1 minute are observable).

For all targets that are detected, we use common measured values for source location, receiver location, array heading, and speed of sound, as well as target-specific time and bearing measurements. These measurements lead to measurement covariance matrices as described in [8]. In addition to target detections, we add a fixed number of false alarm contacts per contact file, uniformly distributed in time-bearing measurement space.

For each run, the ground truth and contact data simulator generates a set of contact files that are processed by both the baseline tracker and the multi-hypothesis tracker, and the measures of performance described above are computed. In addition, the input data and the output tracks are visualized on a simple *x*-*y* MATLAB display. In particular, for each contact file, the measured platform receiver location is displayed in red, target ground truth is displayed in black, the target detection contacts are displayed in green, and the false contacts are displayed in blue. Baseline tracker results are displayed in magenta, and multi-hypothesis tracker results are displayed in red.

3.3 Monte Carlo Study with Benchmark Examples

We consider a number of ground truth scenarios of interest. These scenarios provide relevant benchmark cases for tracker evaluation. We fix simulation parameters as follows.

Platform Ground Truth: The scenarios last for 3600 sec. We have two source/receiver platforms. The first starts at [0, 0] m and travels with constant velocity [0, 5] m/s; the second starts at [1000, 2000] m and travels with constant velocity [0, 3] m/s. Both platforms ping at the start time, and with a 60 sec time repetition interval. These settings lead to 4 contact files per minute, and a total of 240 contact files. The speed of sound is set to 1500 m/s.

Contact Data Quality: The probability of detection is set to 0.5 (or 0.7), and the false alarm rate is 20 per contact file. The measurement error standard deviations are as follows: 10 m for source and receiver *x* and *y* locations, 15 m/s for the speed of sound, 1 deg (or 2 deg) for the array heading, 0.1 sec for the contact time, and 1 deg for the contact bearing.

Tracker Parameters: We use M=3, N=4 for track confirmation, 1 m/s for the prior velocity standard deviation in both *x* and *y*, an association gate parameter of 9.2, which corresponds to a 99% gate, and L=12 for the track classifier threshold, which for these scenarios removes tracks with duration that is roughly three minutes or less. For the baseline tracker, we set K=5 for track termination. For the multi-hypothesis tracker, we consider two values for *N-scan* (0 and 2), and a 90 second track termination time (which, for these scenarios, corresponds to $4 \le K \le 7$).

The target ground truth scenarios that we consider are the following:

- *Fixed clutter point scenario*. A single fixed clutter point at [-20000; 20000] m.
- *Crossing targets scenario*. The first target starts at [20000; -3000] m and has constant velocity [2; 2] m/s; the second target starts at [26000; -3000] m and has constant velocity [-2; 2] m/s.
- *Parallel targets scenario*. The first target starts at [-28000; 3000] m and has constant velocity [0; 2] m/s; the second target starts at [-29000; 3000] m and has constant velocity [0; 2] m/s.
- *Targets near fixed clutter scenario*. The fixed clutter points are at [10000; 10000] m and at [10000; 13000]; the target starts at [9700; 8000] m and has constant velocity [0; 3] m/s.
- *Randomized target motion scenario*. Five targets move according to a nearly constant velocity motion model. For each target, the initial location is chosen randomly within a region centered at [0; 0] with sides of length 60000. The initial velocity is chosen according to a zero mean Gaussian distribution with 1 m/s standard deviation in both *x* and *y*. The process noise parameters are 0.0001 m^2/s^3 in both *x* and *y*. (References [1-2] contain a more detailed discussion of the nearly constant velocity motion model).

For illustration purposes, in Figures 3.1-3.10 we show one realization for each of the five scenarios identified above, with a target probability of detection of 0.7. We also show baseline and multi-hypothesis tracks, where in the latter case we have a hypothesis tree depth of *N-scan*=2.

In Figures 2-3 we see that both trackers are successful in identifying a single fixed clutter track, and no false tracks. Figures 4-5 illustrate the crossing targets scenario. The baseline tracker loses track of one target near the junction, and picks it up afterwards. In the parallel targets scenario illustrated in Figures 6-7, again we see that the baseline tracker has some difficulty in disambiguating the targets; this leads to four tracks rather than two, and increased localization errors. The run in Figures 8-9 illustrates the difficulty of fixed clutter near a moving target, as the baseline tracker loses track on the moving target near the first clutter point. With the randomized target motion run shown in Figures 10-11, both trackers generate six tracks on five targets.

We are interested to quantify the performance of the baseline and multi-hypothesis trackers with the benchmark scenarios that we have illustrated. We fix the platform ground truth, contact data quality, and tracker parameters as discussed previously.

We have five ground truth scenarios of interest. For each choice of detection probability and array heading accuracy

of interest, we generate 10 contact data sets. (Note that each of the contact data sets based on the stochastically generated ground truth scenario is based on a different ground truth realization). For each contact data set, we run the baseline tracker, and we run the multi-hypothesis tracker twice, with two different values for the *N-scan* parameter. Overall performance results (averaged across all runs) are given in Tables 1-4. Note that by *MHT-n*, we mean the multi-hypothesis tracker with *N-scan=n*.

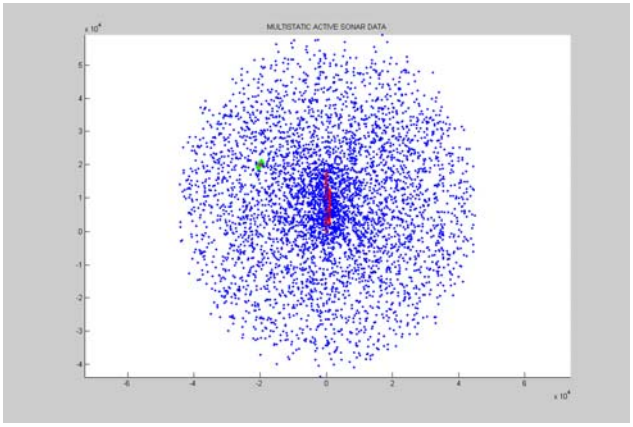

Fig. 2. Fixed clutter point scenario.

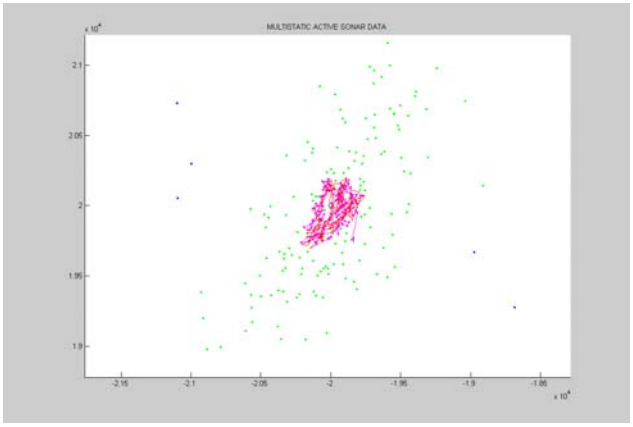

Fig. 3. Close-up view of fixed clutter point scenario.

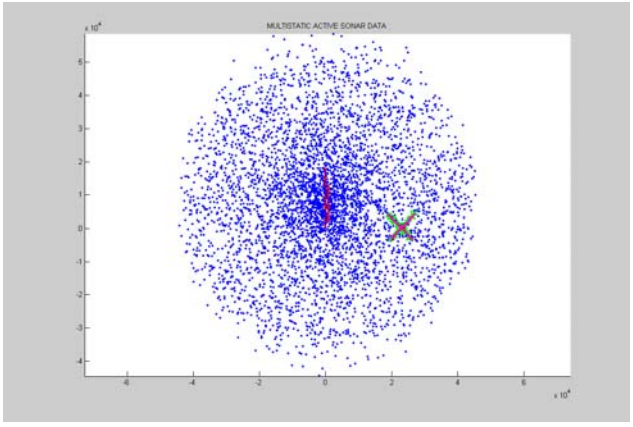

Fig. 4. Crossing targets scenario.

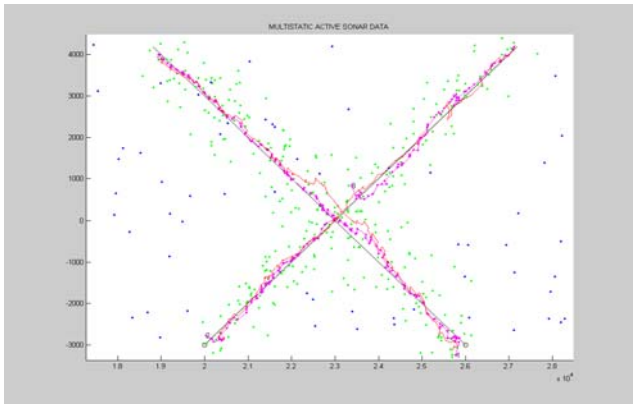

Fig. 5. Close-up view of crossing targets scenario.

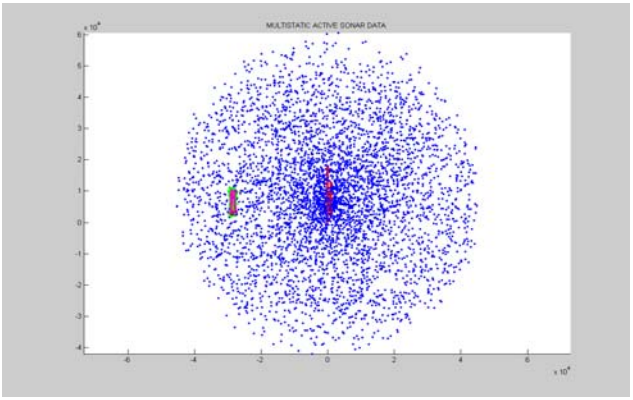

Fig. 6. Parallel targets scenario.

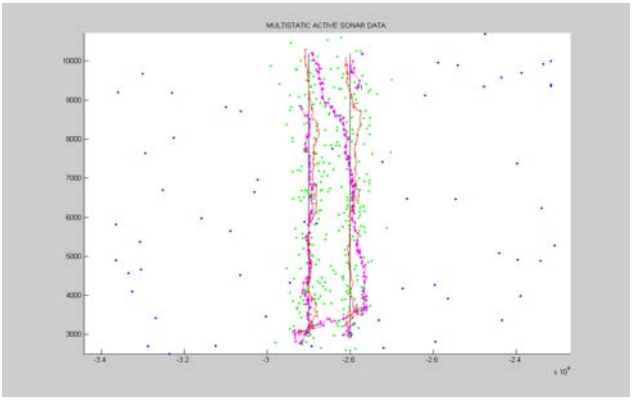

Fig. 7. Close-up view of parallel targets scenario.

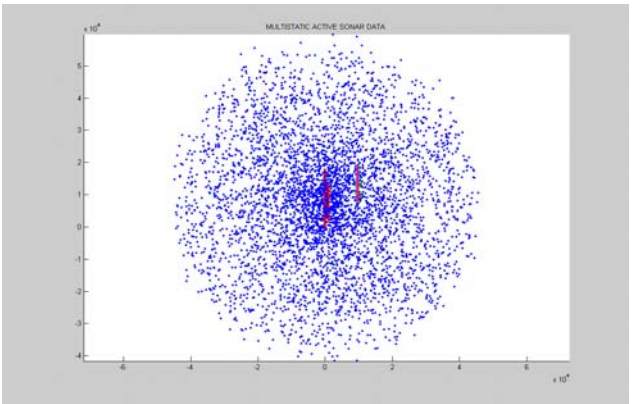

Fig. 8. Targets near fixed clutter scenario.

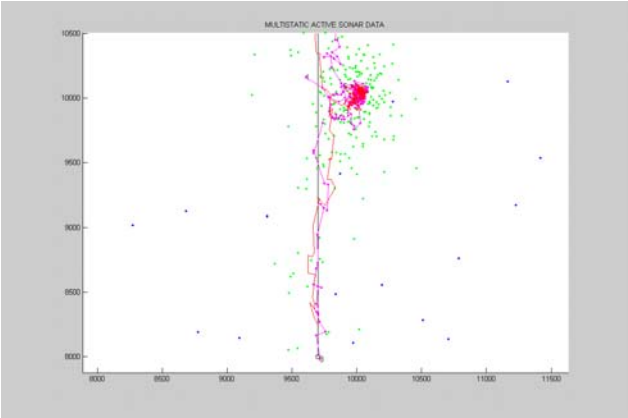

Fig. 9. Close-up view of targets near fixed clutter scenario.

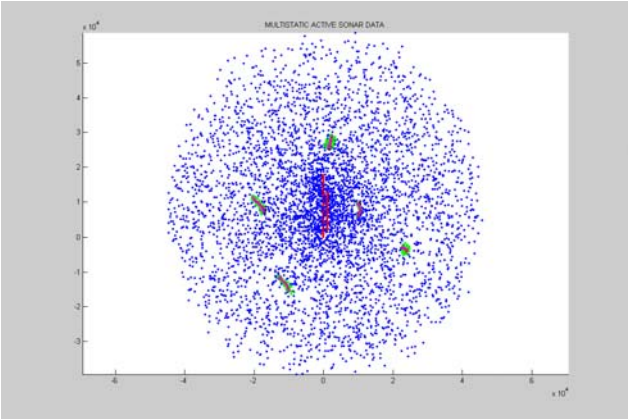

Fig. 10. Randomized target motion scenario.

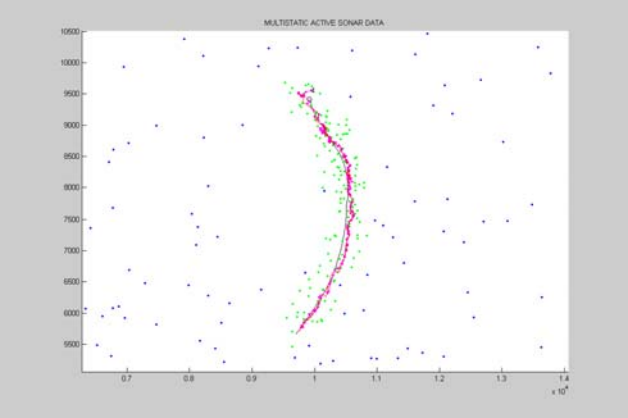

Fig. 11. Close-up view of randomized target motion scenario.

Table 1. Tracker performance (with PD=0.7, heading errors ~ 1 deg).

	PD	FAR	RMSE	T-FRAG
Input	0.700	80 / min	602.97	-
Baseline	0.929	0.02 / hr	230.71	1.61
MHT-0	0.967	0.26 / hr	180.20	1.17
MHT-2	0.969	0.26 / hr	180.97	1.19

Table 2. Tracker performance (with PD=0.5, heading errors ~ 1 deg).

	PD	FAR	RMSE	T-FRAG
Input	0.500	80 / min	616.30	-
Baseline	0.726	0.04 / hr	282.58	2.91
MHT-0	0.869	0.42 / hr	244.11	2.28
MHT-2	0.876	0.38 / hr	243.53	2.23

Table 3. Tracker performance (with PD=0.7, heading errors ~ 2 deg).

	PD	FAR	RMSE	T-FRAG
Input	0.700	80 / min	935.58	-
Baseline	0.928	0.02 / hr	330.31	1.54
MHT-0	0.966	1.16 / hr	279.96	1.14
MHT-2	0.969	1.10 / hr	282.53	1.17

Table 4. Tracker performance (with PD=0.5, heading errors ~ 2 deg).

	PD	FAR	RMSE	T-FRAG
Input	0.500	80 / min	908.44	-
Baseline	0.720	0.02 / hr	436.80	2.89
MHT-0	0.873	1.12 / hr	338.46	2.14
MHT-2	0.878	1.14 / hr	343.94	2.05

We first note the clear benefits of target tracking, with improved target detection, reduced localization errors, and drastic reduction in data volume and false objects, relative to the input data. T-PD consistently exceeds the input PD and the T-RMSE is much lower than the input RMSE. We have not defined the FRAG metric for input data; conceptually, it would be the average number of detections per target. The low T-FRAG values illustrate that the trackers successfully associate contact data and drastically reduce the number of true objects.

We find a clear improvement in performance with the multi-hypothesis tracker, relative to the baseline tracker, in terms of T-PD, T-FRAG, and T-RMSE. On the other hand, the effective partitioning of contact data with the improved data association scheme also leads to the detection of data sequences that occasionally form spurious tracks. Indeed, with our parameter settings, we find a higher T-FAR with the multi-hypothesis tracker than with the baseline tracker.

The improved performance of the multi-hypothesis tracker relative to the baseline tracker is even more significant in harsher tracking conditions. The following more challenging example with lower quality contact data illustrates the potential of the multi-hypothesis tracker. With detection probability of 0.5 and array heading errors of 1 deg, we increase the input FAR to 50 per scan, or 200 per minute. We consider again the single fixed clutter point scenario: for this scenario, which is less challenging than the other scenarios with close targets or targets near clutter points, we had previously observed similar tracking performance between the baseline and multi-hypothesis trackers.

Tracker outputs for the baseline and multi-hypothesis tracker with *N-scan*=2 are illustrated in Figure 12. Unlike Figure 3, here the baseline tracker has difficulty maintaining track on the fixed clutter point, leading to three track fragments. However, the multi-hypothesis tracker successfully maintains track on the clutter point.

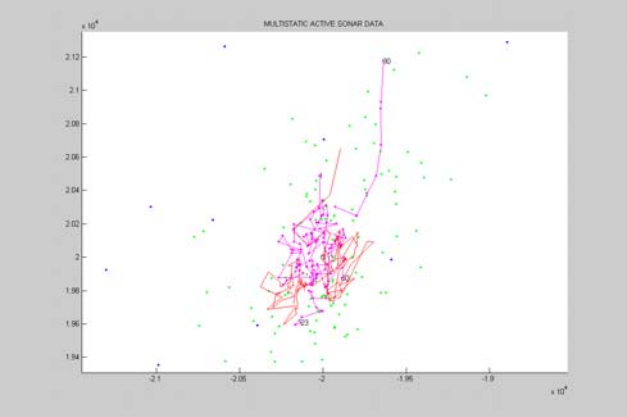

Fig. 12. Close-up view of fixed clutter point tracks for the more challenging detection scenario.

The contact data quality that we have considered in this performance analysis is relatively high. Equally important, the tracking algorithms assume that detection performance and measurement errors are well characterized. In practice, it is difficult to quantify the quality of the contact data. In addition, there may be system timing errors and registration errors, as we find in our analysis of real sea trial data. Thus, the relative performance of the baseline and multi-hypothesis trackers is of interest in more challenging settings. With either tracking algorithm, calibration and registration techniques are necessary components for effective data fusion.

4 Conclusions and Future Work

This paper extends our past work in multistatic sonar tracking algorithms. We introduce a multi-hypothesis multistatic tracking algorithm, which examines numerous contact data association hypotheses and results in improved tracking performance over our earlier baseline algorithm. The multi-hypothesis tracker includes a number of features not generally found in other multi-hypothesis trackers. Most notably, we use an efficient LP-based approach for data association, with a novel modification to improve track continuation of confirmed tracks. The tracker handles system, environmental, and measurement uncertainties by encoding this information in the multistatic contact uncertainty.

There are a number of areas for continued algorithmic development in support of operationally effective multistatic fusion and tracking systems. Variable detection performance and residual system uncertainties and biases significantly impact multi-sensor tracking performance. We plan to develop a two-stage tracking approach in which each source-receiver pair generates its own target tracks, and these are then combined with an automated track fusion algorithm.

In addition, we plan to further exploit information in the contact data. Thus far, besides a simple threshold to remove low-SNR contacts, we have only used information relevant for target localization. The use of simple, single-ping classification algorithms may significantly increase the information content provided to the tracker. Finally, track classification based on kinematic and or feature state estimates may improve the quality of the tracker output.

References

[1] S. Coraluppi and D. Grimmett, Multistatic Sonar Tracking, in Proceedings of the SPIE Conference on Signal Processing, Sensor Fusion, and Target Recognition XII, April 2003, Orlando FL, USA.

[2] S. Coraluppi and D. Grimmett, Multistatic Sonar Tracking, in *Proceedings of the SET Panel Symposium on Target Tracking and Sensor Data Fusion for Military Observation Systems*, October 2003, Budapest, Hungary.

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

[4] Y. Bar-Shalom and X. Li, *Multitarget-Multisensor Tracking*, YBS Publishing, 1995.

[5] L. Stone, C. Barlow, and T. Corwin, *Bayesian Multiple Target Tracking*, Artech House, 1999.

[6] M. van Velzen and R. Laterveer, Measuring Detection Performance in Active Sonar using Object Counting, *NURC SR-331*, La Spezia, Italy, 2001.

[7] S. Jespers and D. Grimmett, Data Association and Interoperability of Multi-static LFAS Platforms, in *Proceedings of the SET Panel Symposium on Multi-Platform Integration of Sensors and Weapon Systems for Maritime Applications*, October 2002, Norfolk VA, USA.

[8] S. Coraluppi, Multistatic Sonar Localization Analysis, *NURC Technical Report SR-377*, June 2003.

[9] 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*, June 2000, San Antonio TX, USA.

[10] P. Storms and F. Spieksma, an LP-based Algorithm for the Data Association Problem in Multitarget Tracking, in *Proceedings of International Conference on Information Fusion*, Paris, France, July 2000.