A Multistage Tracker for Distributed Sensor Fields

Evan Hanusa and David W. Krout
Applied Physics Laboratory
University of Washington
1013 NE 40th St.
Seattle, WA 98105-6698
Email: {hanusaem,dkrout}@apl.washington.edu

Abstract— **This paper presents the results of using a multiple stage tracker to improve tracking results on a multistatic sonar dataset. The tracker consists of a predetection fusion step, an extended Kalman filter implementation of joint probabilistic data association (JPDA), and a Monte Carlo implementation of JPDA. The predetection fusion step, combined with the first EKF-JPDA step is used to detect targets and initialize tracks in the Monte Carlo JPDA tracker. The Monte Carlo JPDA tracker allows for the use of multiple models, as well as accurately modeling the measurement uncertainty without linearity approximations. The multiple stage tracking system results in improved localization and decreased fragmentation when compared to the baseline tracker.**

I. Introduction

Multistatic active sonar fields are becoming more common in marine surveillance applications. The large number of receivers and transmitters allows for increased (overall) probability of detection and coverage area. However, the increased number of sensors also increases the number of clutter contacts, which can make tracking dim objects difficult. Here we evaluate a multi-stage tracking framework that cascades two trackers with the goal of accurately estimating object locations in a multistatic field.

The complete tracking system consists of several steps: a clustering-based predetection fusion, a extended Kalman filter (EKF) with *joint probabilistic data association* (JPDA) for cluster-track association, and a multiple model *Monte Carlo JPDA* (MCJPDA) implementation of a particle filter. The clustering-based predetection fusion step clusters all the contacts across receivers for a single transmitted ping. The clusters are fused and passed to the EKF-JPDA filter. This tracker stage manages all potential tracks and passes confirmed tracks (and their initial state estimates) to the MCJPDA tracker stage. The MCJPDA tracker stage improves tracking performance in both localization and fragmentation.

The overall tracking performance is evaluated on the PACsim dataset, a simulated multistatic dataset [1]. Two of the dataset's three scenarios are used to evaluate the tracker. Each has five transmitters, 16 receivers. Both fixed and mobile targets are modeled. It includes objects of varying probability of detection (P_d), including very dim targets, characterized by a P_d of less than 0.1 per receiver.

Section II discusses other approaches to tracking multistatic sonar data. Section III details the different stages of the tracker. Section IV describes the PACsim dataset used to evaluate the tracker. Section V discusses the tracking results and compares them to the baseline.

II. Related Work

As multistatic sensor systems have become popular, many approaches to tracking have been explored applied to the problem. One approach uses multiple levels of Maximum Likelihood *Multiple Hypothesis Tracking* (MHT) to detect and track targets in a multistatic field [2]. The first pass detects targets across all receivers on a single transmitted ping, the second tracks targets over time. In the ML-MHT implementation, only contacts are passed between the receivers.

Other approaches apply existing trackers to the raw data. A Gaussian mixture implementation of MHT, combined with careful analysis of the data and parameter tuning is able to attain good tracking results. It has also been modified to include feature data [3]. Two more computational intensive trackers, the *Sequentially Sampled Particle Filter with Existence* (SSPFE) and *Joint Integrated PDA* (JIPDA) trackers were able to track dim targets in moderate clutter, but struggled in higher clutter scenarios [4].

An alternative to the trackers above is to use a *predetection fusion* step before tracking. The goal of predetection fusion is to combine measurements across receivers to reject clutter, decrease the number of contacts presented to the tracker, and improve localization. Predetection fusion has shown good performance on multistatic tracking datasets [5], [6]. The clustering-based predetection fusion is used as the first stage in the tracker presented in this work.

III. Tracking Framework

The multistage tracker presented here consists of a clustering-based predetection fusion step, an EKF-based implementation of JPDA, and a Monte Carlo implementation of JPDA. Predetection fusion and the EKF-JPDA tracker are used to initialize tracks and reject clutter contacts, while the more computationally intensive MCJPDA tracker is used to obtain an accurate estimate of target locations.

A. Clustering-Based Predetection Fusion

As mentioned in Section I, there are many difficulties in tracking targets in multistatic active sonar fields. Many standard approaches are able to track targets with some success, but performance often suffers on very dim or maneuvering

978-0-933957-40-4 ©2013 MTS

This is a DRAFT. As such it may not be cited in other works.
The citable Proceedings of the Conference will be published in
IEEE Xplore shortly after the conclusion of the conference.

targets. Results can be improved through careful tuning of the tracking parameters and post-track classification [3], [4], [7].

Another approach is to apply a fusion step before tracking the data, referred to as predetection fusion. This approach combines the sensor measurements before thresholding or tracking. Here, the clustering-based predetection fusion approach is applied to the raw data [6].

Most clustering algorithms operate on a similarity (or distance) matrix that is created by calculating the pairwise distance between every measurement. In multistatic sonar, the choice of distance measure is key, and must appropriately model the measurement error distributions. The likelihood-based similarity used for clustering allows for comparison of measurements across receivers, as well as incorporation of Doppler, amplitude, and other features [6].

The similarity matrix is used as the input to agglomerative clustering, with "complete" similarity used as the inter-cluster similarity [8]. Complete similarity defines the similarity of two clusters as the similarity of the two *least* similar contacts in the clusters. This results in more, smaller clusters and has shown to have good performance in clustering sonar contacts [9].

After clustering, the contacts in each cluster are fused into a single measurement, and can be classified as based on characteristics of the clusters themselves (e.g. number of contacts and posterior probability of the fused contact) or features of the contacts within the cluster (e.g. averaged features).

B. Joint Probabilistic Data Association with Extended Kalman Filter

After the contacts have been clustered and fused in the predetection fusion step, they are passed to an EKF implementation of JPDA. The goal of this stage is twofold: initialization of target tracks for the next stage and selection of contacts to pass to the next stage. Both of these goals result in decreased runtime for the computationally intensive MCJPDA tracker.

As there are many references for JPDA, the derivation is not included here. For an in-depth discussion of JPDA, the reader is directed to [10], [11].

The motion model used for the extended Kalman filter is a nearly constant velocity model, and therefore has states $[x, y, \dot{x}, \dot{y}]$. The process noise weight is $\sigma_q = 1 \times 10^{-3}$. P_d for FM and CW pings are selected to be 0.7 and 0.2, respectively. These differ from the per-receiver values for P_d, as they are the detection probabilities at the output of the predetection fusion stage.

Results using only the predetection fusion and EKF-JPDA tracker have been presented previously, and will be used as a baseline [9], [6].

1) Track Management: Track initiation is done using a 1-point initiation technique. Tracks are started at all clusters that do not gate with an existing track, and are initialized to have a velocity of zero.

Track confirmation and deletion are handled by a slightly-modified *Sequential Probability Ratio Test* (SPRT). The standard SPRT is modified to have a maximum score, which stops high-score tracks that lose the target from continuing for too long.

C. Monte Carlo Joint Probabilistic Data Association with Multiple Motion Models

The EKF-JPDA implementation suffers for two main reasons: the Kalman filter is unable to track targets as they make a sharp turn and the modeling of the error from the fused clusters is inaccurate. To overcome these, the second stage of the tracker is an implementation of MCJPDA, modified to include multiple models [12]. MCJPDA allows for more accurate tracking, however it is much more computationally complex. To ensure that resources were allocated effectively, MCJPDA tracks were only initialized on tracks that had been confirmed by the EKF-JPDA. Additional logic ensures that tracks are not duplicated in the MCJPDA tracker, which is a possibility due to the tendency of the EKF-JPDA tracker to fragment tracks.

The inputs to the MCJPDA filter are the raw measurements, not the fused clusters. This allows the algorithm to appropriately model the measurement error in bearing, range, and Doppler.

Two models are used in this tracker: one with a low process noise ($\sigma_q = 5 \times 10^{-7}$) and one with a high process noise ($\sigma_q = 1 \times 10^{-3}$). The first model allows for accurate tracking while the target is not maneuvering, and the second allows for the target to be tracked if it makes a sharp turn.

MCJPDA was implemented using the following parameters and design choices. Each track had 1000 particles, and resampling was done if the number of effective particles dropped below 500. The proposal density used was a mixture of the bootstrap density (the distribution due to the dynamics of the target) and the measurements which gated with the prior estimate of target position, as recommended in [12]. P_d for FM and CW pings are selected to be .25 and .05, respectively. These values were selected by calculating the average P_d for targets in the dataset.

IV. PACSIM DATASET

The PACsim dataset was created by Doug Grimmett for the MSTWG (*Multistatic Tracking Working Group*) [1]. The dataset consists of three scenarios, each with 5 transmitters and 16 receivers. The first two scenarios are used to evaluate the cascaded tracker system Both *continuous wave* (CW) and *frequency modulated* (FM) waveforms were simulated, with the transmitted pings alternating between the two. For CW and FM pings, the bearing measurement error is additive Gaussian with mean $0.0°$ and standard deviation $4.0°$. For CW pings, the bistatic range error is additive Gaussian with mean 0.0 s and standard deviation 0.1 s. Additionally, a Doppler measurement is available, and it is corrupted with Gaussian noise with standard deviation 0.21 m/s. For FM pings, the bistatic range error is additive Gaussian with mean 0.0 s and standard deviation 0.4 s. The soundspeed is a constant 1500 m/s and there is no blanking region included in the simulation. Scenarios A and B have a clutter rate of approximately 22

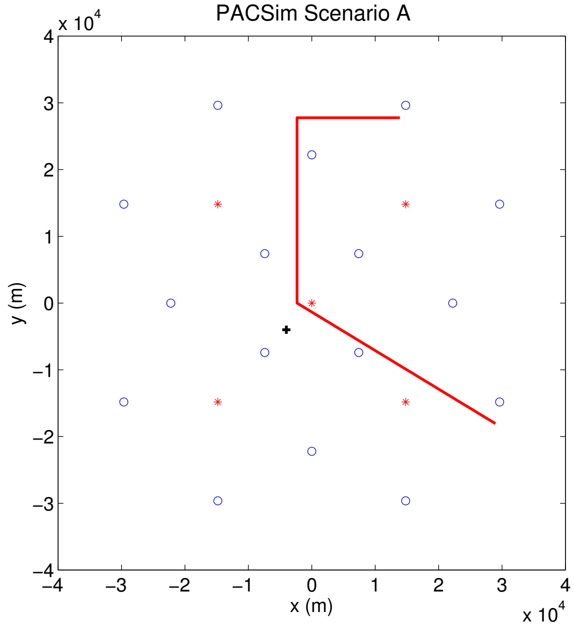

Fig. 1. Ground truth plot and sensor layout for Scenario A of the PACsim data set. Two targets are present. One moves during the scenario (red line), and the other is stationary (black '+'). There are 5 sources (red stars) and 16 receivers (blue circles).

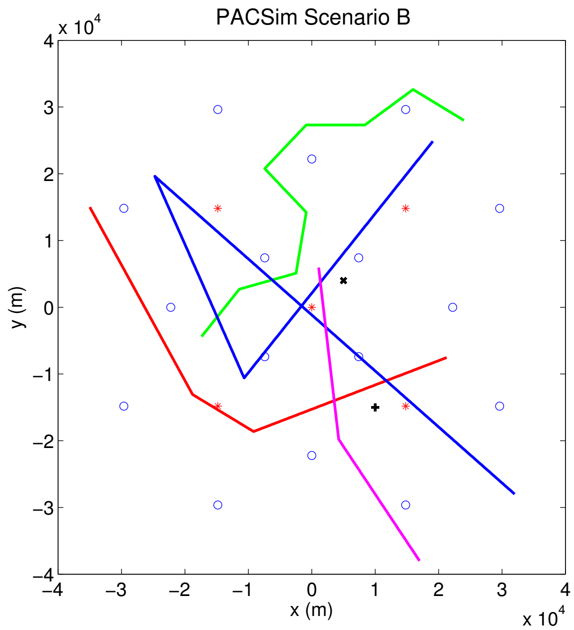

Fig. 2. Ground truth plot and sensor layout for Scenario B of the PACsim data set. Six targets are present. Four move during the scenario (red, blue, green and magenta lines), and the other two are stationary (black '+' and 'x'). There are 5 sources (red stars) and 16 receivers (blue circles).

false contacts per receiver per ping. Figures 1 and 2 show the ground truth for the scenarios included with the PACsim dataset.

The PACsim dataset contains targets of varying difficulty. For scenario A, there is one moving target with a moderate probability of detection, P_d and one fixed clutter point. Scenario B has two targets with moderate P_d, two difficult targets and two fixed clutter points. Table I compares the probability of detection across scenarios and the type of ping (CW vs FM).

	Target	1	2	3	4	5	6
A	CW	0.21	0				
A	FM	0.47	0.44				
B	CW	0.14	0.15	0.10	0.01	0	0
B	FM	0.53	0.52	0.11	0.11	0.47	0.47

TABLE I

COMPARISON OF PROBABILITY OF DETECTION ACROSS SCENARIOS AND TRANSMITTED WAVEFORM TYPE.

V. RESULTS

Here we discuss the results of the cascaded tracking system. The baseline used is the JPDA tracker with extended Kalman filter for state estimation and prediction, using a clustering-based predetection fusion step.

A. EKF-JPDA Baseline

In scenario A, there are two targets. A fixed target, which is tracked well by the baseline tracker. It also contains a moving target, that moves in straight lines for a long period of time, then makes sharp turns. As shown in Figure 3, the tracks tend to fragment on the sharp corners, or at the least overshoot. This is the result of the mismatch between the nearly constant velocity model with the target dynamics. It would be possible to increase the process noise to allow the target to be tracked as it turns, however that would increase the localization error in the straight sections.

PACsim scenario B contains six targets. There are two fixed targets, which are tracked well by the baseline tracker. The two high P_d targets (red, green) are well localized, but again suffer from overshoot on the turns as well as fragmentation. The lower P_d targets (blue, magenta) are tracked well for portions of the scenario, but are difficult to track due to their very low probability of detection.

B. Cascaded Trackers

Using the cascaded tracking system, the target is tracked continuously around the turns, with minimal overshoot. This is due to both the multiple models used for tracking and also the accurate modeling of the measurement uncertainty. The result is improved localization error in the straight sections, as well as decreased fragmentation. There is a small increase in localization error for the fixed target on some pings. This is due to the "turn" model matching contacts that are far away from the fixed clutter point. This could be avoided by automatically classifying the target as a fixed point, and disabling the high-variance turn model.

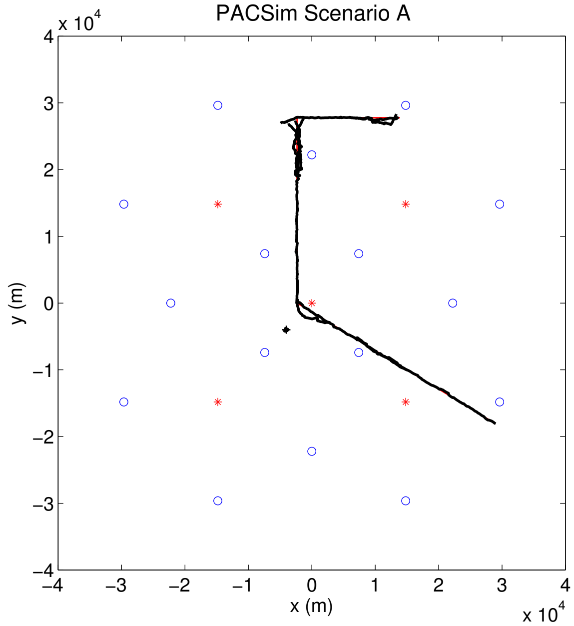

Fig. 3. Results of the clustering preprocessing and EKF-based tracker on scenario A. Note the overshoot and track break on the sharp turns.

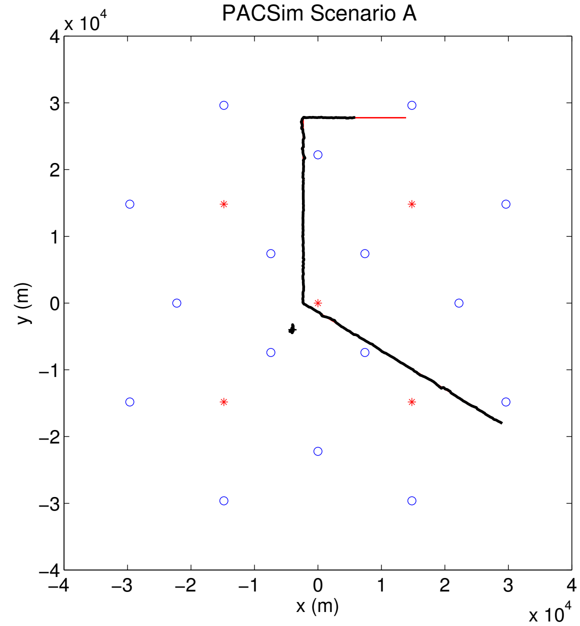

Fig. 5. Results of the cascaded tracking system. Note the lack of fragmentation and overshoot on the turns.

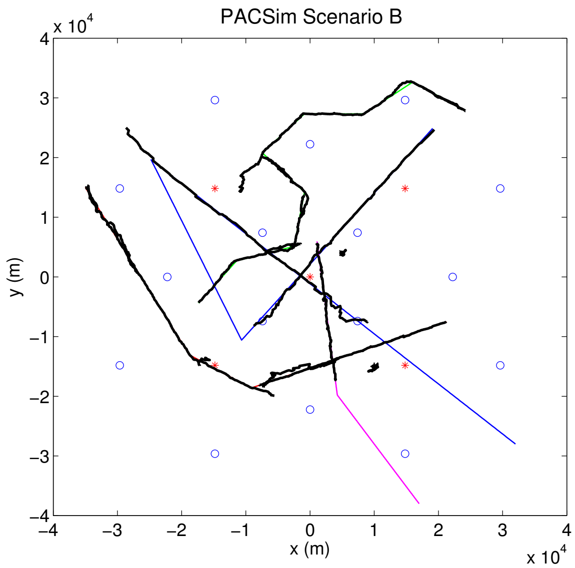

Fig. 4. Results of the clustering preprocessing and EKF-based tracker on scenario B. Note the multiple parallel tracks that exist until they converge and are appropriately merged. The EKF tracker again performs poorly on the turns, resulting in fragmentation and increased error.

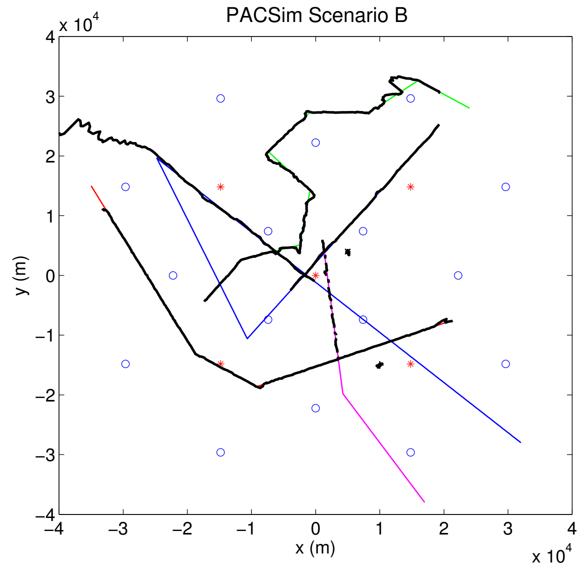

Fig. 6. Results of the cascaded tracking system. The cascaded tracker reduces fragmentation for the green and red targets. It also improves localization for the red target. For the lower P_d targets (blue, magenta), the performance is very similar to the baseline. It has worse performance in some areas, and improved in others (near the end of the blue track, for example).

VI. Conclusion

Using the cascaded tracker in PACsim scenario A results in obvious gains. Localization error and track fragmentation are both decreased. Performance on PACsim B is improved for high P_d targets. However, results are mixed for the lower P_d targets. Some areas show improvements, while others are worse. This is likely due to the incorrect selection of P_d parameters for the data. The same P_d is used for all tracks, and it would be better to use location-dependent or track-dependent values.

There are several open questions that the authors are currently exploring. There are many ways to have the trackers interact, and it is not yet known which is the best. For example, if a contact gates with an MCJPDA track, it could be excluded from the EKF tracker. This could potentially allow for the detection of closely spaced targets. For the MCJPDA tracker, only two models are used: a high and low noise model. Another option is to use turn models, which may perform better for other types of data.

The EKF-JPDA tracker is only one method for initializing tracks for MCJPDA. The use of a particle filter has been explored for similar data [13]. Another possible method is to extend the clustering step across multiple pings, and use that method for initializing tracks.

One research area of interest is applying this type of cascaded tracker to multstatic passive radar data, similar to the approach used in single frequency networks [14].

Acknowledgment

This work was funded by the U.S. Office of Naval Research, Contract Number N00014-07G-0557, Delivery Order #28.

References

[1] D. Grimmett, "MSTWG - PACsim dataset." SPAWAR Systems Center Pacific, Released to the MSTWG, Powerpoint, July 2011.

[2] S. Coraluppi and C. Carthel, "An ML-MHT approach to tracking dim targets in large sensor networks," in *Information Fusion (FUSION), 2010. 13th International Conference on*, July 2010.

[3] G. Mellema, "Feature-assisted multistatic tracking using the PACsim data set," in *Information Fusion (FUSION), 2012 15th International Conference on*. IEEE, 2012, pp. 2608–2615.

[4] F. Fletcher and S. Arulampalam, "A comparison of existence-based multitarget trackers for multistatic sonar," in *Information Fusion (FUSION), 2012 15th International Conference on*. IEEE, 2012, pp. 2362–2369.

[5] R. Georgescu and P. Willett, "Predetection fusion with Doppler measurements and amplitude information," *Oceanic Engineering, IEEE Journal of*, vol. 37, no. 1, pp. 56–65, 2012.

[6] E. Hanusa, D. Krout, and M. Gupta, "Contact clustering and fusion for preprocessing multistatic active sonar data," in *Information Fusion (FUSION), 2013 16th International Conference on*. IEEE, 2012.

[7] D. Grimmett, C. Wakayama, and R. Plate, "Multistatic post-track classification using a target strength function," in *Information Fusion (FUSION), 2012 15th International Conference on*. IEEE, 2012, pp. 2339–2346.

[8] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer, 2005, vol. 27, no. 2.

[9] E. Hanusa, "Incorporation of features in multistatic active sonar tracking," Ph.D. dissertation, University of Washington, Seattle, December 2012.

[10] Y. Bar-Shalom and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE Journal of Oceanic Engineering*, vol. 8, no. 3, pp. 173–184, 1983.

[11] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA: Artech House, 1999.

[12] J. Vermaak, S. Godsill, and P. Perez, "Monte carlo filtering for multi-target tracking and data association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 309–332, January 2005.

[13] J. Georgy, A. Noureldin, and G. R. Mellema, "Clustered mixture particle filter for underwater multitarget tracking in multistatic active sonobuoy systems," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 42, no. 4, pp. 547–560, 2012.

[14] M. Daun, U. Nickel, and W. Koch, "Tracking in multistatic passive radar systems using DAB/DVB-T illumination," *Signal Processing*, vol. 92, no. 6, pp. 1365–1386, 2012.