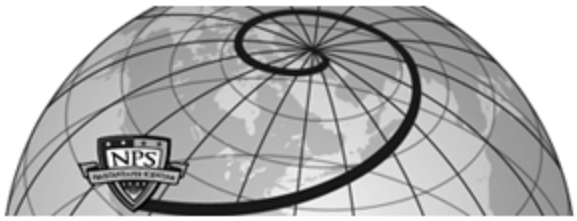


1998-01

# Evaluation and extensions of the probabilistic multi-hypothesis tracking algorithm to cluttered environments

Hutchins, Robert G.

Monterey, California. Naval Postgraduate School



NPS-EC-98-015

# NAVAL **POSTGRADUATE SCHOOL**
## Monterey, California




**Evaluation and Extensions of the Probabilistic Multi-Hypothesis Tracking Algorithm to Cluttered Environments**

by

R. *G.* Hutchins and D. **T.** Dunham

January 1998



Prepared for: Naval Undersea Warfare Center, Newport, RI

**NAVAL POSTGRADUATE SCHOOL**
**Monterey, California**

RADM ROBERT C. CHAPLIN
Superintendent

R. Elster
Provost

This report was sponsored by the Naval Undersea Warfare Center, Newport, RI.

Approved for public release; distribution is unlimited.

The report was prepared by:

[redacted]

R. G. HUTCHINS
Associate Professor
Department of Electrical and
Computer Engineering

Reviewed by:

[redacted]

JEFFREY B. KNORR
Chairman
Department of Electrical and
Computer Engineering

Released by:

[redacted]

DAVID W. NETZER
Associate Provost and
Dean of Research

# REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188



| . AGENCY USE ONLY (Leave blank) | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | January 1998 | Final Rep[illegible]t |

1 TITLE AND SUBTITLE
Evaluation and Extensions of the Probabilistic Multi-Hypothesis Tracking Algorithm to Cluttered Environments

5. FUNDING NUMBERS
N6660498WR80335

6. AUTHOR(S)
R.G. Hutchins and D.T. Dunham

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Department of Electrical and Computer Engineering
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER
NPS-EC-98-015

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Naval Undersea Warfare Center
1176 Howell Street
Newport, RI 02841-5047

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

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

12b. DISTRIBUTION CODE
A

13. ABSTRACT (Maximum 200 words)

This research examines the probabilistic multi-hypothesis tracker (PHMT), a batch-mode, empirical, Bayesian data association and tracking algorithm. Like a traditional multihypothesis tracker (MHT), track estimation is deferred until more conclusive data is gathered. However, unlike a traditional algorithm, PMHT does not attempt to enumerate all possible combinations of feasible data association links, but uses a probabilistic structure derived using expectation-maximization. This study focuses on two issues: the behavior of the PMHT algorithm in clutter and algorithm initialization in clutter. We also compare performance between this algorithm and other algorithms, including a nearest neighbor tracker, a probabilistic data association filter (PDAF), and a traditional measurement-orientedMHT algorithm.

14. SUBJECT TERMS
Kalman filtering, multihypothesistracking, multitarget tracking in clutter

47

16. PRICE CODE
16. PRICE CODE

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OFABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | SAR |

# Evaluation and Extensions of the Probabilistic Multi-Hypothesis Tracking Algorithm to Cluttered Environments

R.G. Hutchins and D.T. Dunham

## 1. Introduction

Dr. Roy Streit and colleagues at the Naval Undersea Warfare Center, Division Newport, have developed a probabilistic multi-hypothesis tracking (PMHT) algorithm that simplifies multi-hypothesis tracking, potentially extending the applicability of these techniques to a broader range of problems. The algorithm does not contain an inherent initialization strategy; nor has there been a realistic study of the behavior of the algorithm in cluttered environments. The purpose of the research detailed in this report has been four-fold: to test and validate this new algorithm by comparing it with a nearest neighbor algorithm, a traditional multiple hypothesis tracking (MHT) algorithm, and a probabilistic data association filter (PDAF) using standardized test scenarios; to study comparative algorithm performance in the presence of clutter; to evaluate system performance using realistic initiation procedures in the presence of clutter, and to initiate a study of attribute-augmented measurement-to-track association procedures. The ultimate goal is to develop a workable set of algorithms that is practical and that will achieve reasonable performance in the presence of clutter.

This report is divided into the following sections: Section 2 discusses the specific algorithms and equations used in this research, including the baseline PMHT algorithm, its extensions and alternative initialization strategies, the MHT algorithm and PDAF algorithm **as** they have been implemented in this research, and the attribute-augmented model studied in this initial research. Section 3 presents the specific scenarios used in the comparative algorithm analysis, along with the results of this analysis. Section **4** gives our summary and conclusions.

## 2. Algorithms and Equations

The specific algorithms and equations used in this research are detailed in this section. These algorithms include the baseline PMHT and its extensions, the MHT, the PDAF, and algorithms pertaining to measurement processing and track initiation. The subsections below begin with assumed target motion models and measurement processing, which are used by all algorithms tested in the scenarios detailed in Section **3,** followed by track initiation, a detailed discussion of the PMHT and its extensions **as** implemented here, the nearest neighbor and PDAF algorithms, and the MHT. This section concludes with a discussion of attribute data **as** implemented in this study.

### 2.1 Target Motion Model

Straight line target motion is assumed, with perturbations given by additive, Gaussian noise. Hence, a real target is assumed by the algorithm to move according to the equation:

$$\mathbf{x}(t,k+1) \equiv \begin{pmatrix} x(t,k+1) \\ v_x(t,k+1) \\ y(t,k+1) \\ v_y(t,k+1) \end{pmatrix} = \begin{bmatrix} 1 & \Delta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x(t,k) \\ v_x(t,k) \\ y(t,k) \\ v_y(t,k) \end{pmatrix} + \begin{pmatrix} w_1(t,k) \\ w_2(t,k) \\ w_3(t,k) \\ w_4(t,k) \end{pmatrix}, \text{ or} \qquad (2.1)$$

$$\mathbf{x}(t,k+1) = \mathbf{A}(\Delta)\mathbf{x}(t,k) + \mathbf{w}(t,k)$$

where $x(t,k)$ and $v_x(t,k)$ are the position and velocity components of target t at time k in the x dimension, $y(t,k)$ and $v_y(t,k)$ are similar quantities in the y dimension, Ais the time between measurement scans, and $w_i(t,k)$ are noise components with covariance given by:

$$Q(t,k) = q\begin{bmatrix} \Delta^3/3 & \Delta^2/2 & 0 & 0 \\ \Delta^2/2 & \Delta & 0 & 0 \\ 0 & 0 & \Delta^3/3 & \Delta^2/2 \\ 0 & 0 & \Delta^2/2 & \Delta \end{bmatrix} = Q$$

and $q$ is a parameter that captures the wideband behavior associated with target maneuvers. In practice, q is used to adjust the performance of the Kalman Filter. When $q$ is zero there is no plant noise, and the filtering/smoothing algorithms attempt to fit an exact straight line to the data. However, this can lead to covariance collapse in the Kalman algorithms, giving nearly singular covariance matrices. Hence, $q$ is usually set to a small but nonzero value when non-maneuvering targets are to be tracked. When target maneuvers are expected, $q$ can be set to a value consistent with the expected frequency and volatility of the maneuvers.

## 2.2 Measurement Processing

The underlying measurement model assumed in all work covered in this research is that of a sensor that returns range and bearing information corrupted by additive, gaussian noise. Hence, the underlying target measurement is of the form:

$$\begin{pmatrix} r(m,k) \\ \phi(m,k) \end{pmatrix} = \begin{pmatrix} r_a(t,k) \\ \phi_a(t,k) \end{pmatrix} + \begin{pmatrix} e_r(t,m,k) \\ e_\phi(t,m,k) \end{pmatrix}$$

where $r(m,k)$ is the measured range for measurement $m$ at time $k$ (assuming the measurement comes from target $t$), $\phi(m,k)$ is the measured bearing for measurement $m$ at time $k$ (assuming the measurement comes from target $t$), $r_a$ and $\phi_a$ are the actual (true) values of these quantities, and $e_r$ and $e_\phi$ are the errors in these quantities, where $e_r$ and $e_\phi$ are normally distributed random variables with zero mean and joint covariance matrix given by:

$$\begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\phi^2 \end{bmatrix}$$

The underlying methodology used to combine measurements into track estimates is the Kalman Filter or its related algorithm, the Kalman Smoother. The Kalman methodology requires a linear state equation (discussed above) and a linear measurement equation. The above state equation for a non-maneuvering target is linear in Cartesian coordinates, but the bearing measurements are nonlinear in the Cartesian state coordinates. Lerro and Bar-Shalom have demonstrated that preprocessing range-bearing measurements into Cartesian space prior to implementing the Kalman algorithm is superior to utilizing the raw range-bearing measurements and employing the Extended Kalman Filter or Smoother [1]. They recommend use of the following 'debiased' equations to convert the range-bearing measurements into Cartesian coordinates:

$$\mathbf{z}(m,k) = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r\cos(\phi)[1-(e^{-\sigma_\phi^2} - e^{-\sigma_\phi^2}/2)] \\ r\sin(\phi)[1-(e^{-\sigma_\phi^2} - e^{-\sigma_\phi^2}/2)] \end{pmatrix} \qquad (2.2)$$

where $x = x(m,k)$, $y = y(m,k)$, $r = r(m,k)$, $\phi = \phi(m,k)$, the Cartesian and polar coordinate versions of measurement $m$ at time $k$. The corresponding covariance matrix is given by:

$$R(m,k) = \begin{bmatrix} R^{11} & R^{12} \\ R^{12} & R^{22} \end{bmatrix} \qquad (2.3)$$

where

$$
\begin{aligned}
R^{11} = \; & r^2 e^{-2\sigma_\phi^2}[\cos^2(\phi)(\cosh(2\sigma_\phi^2) - \cosh(\sigma_\phi^2)) \\
& + \sin^2(\phi)(\sinh(2\sigma_\phi^2) - \sinh(\sigma_\phi^2))] \\
& + \sigma_r^2 e^{-2\sigma_\phi^2}[\cos^2(\phi)(2\cosh(2\sigma_\phi^2) - \cosh(\sigma_\phi^2)) \\
& + \sin^2(\phi)(2\sinh(2\sigma_\phi^2) - \sinh(\sigma_\phi^2))] \\
R^{22} = \; & r^2 e^{-2\sigma_\phi^2}[\sin^2(\phi)(\cosh(2\sigma_\phi^2) - \cosh(\sigma_\phi^2)) \\
& + \cos^2(\phi)(2\sinh(2\sigma_\phi^2) - \sinh(\sigma_\phi^2))] \\
& + \sigma_r^2 e^{-2\sigma_\phi^2}[\sin^2(\$)(2\cosh(2\sigma_\phi^2) - \cosh(\sigma_\phi^2)) \\
& + \cos^2(\phi)(2\sinh(2\sigma_\phi^2) - \sinh(\sigma_\phi^2))] \\
R^{12} = \; & \sin(\phi)\cos(\phi)e^{-4\sigma_\phi^2}[\sigma_r^2 + (r^2 + \sigma_r^2)(1 - e^{-\sigma_\phi^2})]
\end{aligned}
$$

In this approach, each Cartesian measurement at each time has a distinct set of measured values and a distinct covariance. In this research we assume that measurements arrive in scans with multiple measurements in each scan, and each measurement in a given scan arrives at exactly the same time. The time between scans is assumed constant.

The above assumptions and equations result in **a** linear measurement model for the processed measurements:

$$
\mathbf{z}(t,k) \equiv \begin{pmatrix} x(t,k) \\ y(t,k) \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} \mathrm{x}(t,k) \\ \mathrm{vx}(t,k) \\ \mathrm{y}(t,k) \\ \mathrm{vy}(t,k) \end{pmatrix} + \begin{pmatrix} \varepsilon_1(m,k) \\ \varepsilon_2(m,k) \end{pmatrix} \equiv \mathbf{H}\mathbf{x}(t,k) + \varepsilon_{m,k} \qquad (2.4)
$$

where $\varepsilon_{m,k}$ is two-dimensional Gaussian noise with the covariance given by $\mathbf{R}(m,k)$. This model, combined with the linear state equation above, provide the matrices and assumptions required by the Kalman filtering/smoothing algorithms used throughout this research. **A** somewhat inconvenient feature of this approach is that each measurement at each scan time has a different measurement covariance matrix. This feature requires some modification to other algorithms, as discussed below.

Most of the results detailed in section 3 below use the debiased conversion measurement algorithm given in equations 2.2-2.4. However, some comparison runs were made on the PMHT algorithm with a competing Extended Kalman Smoother used in place of the debiased conversion processing discussed above. These results are also reported in section 3.

## 2.3 Baseline Probabilistic Multi-Hypothesis Tracker Algorithm

The algorithm is taken from Streit and Luginbuhl [2,3], where the linear Gaussian case is assumed for actual targets (Section 5 of [2]). Straight line target motion is assumed for actual target tracks, along with the linear measurement model discussed above.

The baseline algorithm consists of an initialization step, where an initial sequence of target state estimates and target measurement probabilities is specified for each target at each measurement scan time:

$$
\begin{aligned}
\hat{\mathbf{X}}_t^{(0)} &= \left\{ \hat{\mathbf{x}}_{t,0}^{(0)} \;\; \hat{\mathbf{x}}_{t,1}^{(0)} \;\; \ldots \;\; \hat{\mathbf{x}}_{t,T}^{(0)} \right\} \\
\mathbf{P}_t^{(0)} &= \left\{ \mathbf{P}_{t,0}^{(0)} \;\; \mathbf{P}_{t,1}^{(0)} \;\; \ldots \;\; \mathbf{P}_{t,T}^{(0)} \right\} \qquad (2.5) \\
\Pi_t^{(0)} &= \left\{ \pi_{t,0}^{(0)} \;\; \pi_{t,1}^{(0)} \;\; \ldots \;\; \pi_{t,T}^{(0)} \right\}, \text{ with } \pi_{t,k}^{(0)} > 0
\end{aligned}
$$

Here, t specifies the target model (t = 1,..,M), k specifies the time (k = 0,...,T), m specifies the measurement at time k (m = 1,..,$n_k$), and the superscript i (= 0 above for initialization) indicates the algorithm iteration number. **Also,** the matrix $\mathbf{P}_{t,k}^{(i)}$ is the covariance associated with the smoothed state estimate $\hat{\mathbf{x}}_{t,k}^{(i)}$ for any iteration i **of** the PMHT. The values $\pi_{t,k}^{(i)}$ specify the estimated probability that **a** measurement at scan time k is assigned to target model t after i iterations of the algorithm. At each iteration, the set of assignment weights is computed using:

$$\omega_{t,m,k}^{*(i+1)} = N(\mathbf{z}_{m,k}|\hat{\mathbf{z}}_{t,k}^{(i)}, \mathbf{P}_{t,k}^{(i)}, \mathbf{R}_{m,k}) \text{ for a target track,} \tag{2.6}$$

$$\omega_{t,m,k}^{*(i+1)} = \rho \text{ (an adjustable parameter) for clutter, and} \tag{2.7}$$

$$\omega_{t,m,k}^{(i+1)} = \frac{\omega_{t,m,k}^{*(i+1)}}{\sum_{s=1}^{M} \pi_{s,k}^{(i)} \omega_{s,m,k}^{*(i+1)}}, \text{ where} \tag{2.8}$$

$$N(\mathbf{z}_{m,k}|\hat{\mathbf{z}}_{t,k}^{(i)}, \mathbf{P}_{t,k}^{(i)}, \mathbf{R}_{m,k}) = \frac{\mathrm{Exp}[-\frac{1}{2}(\mathbf{z}_{m,k} - \hat{\mathbf{z}}_{t,k}^{(i)})^{\mathrm{T}} \Sigma^{-1} (\mathbf{z}_{m,k} - \hat{\mathbf{z}}_{t,k}^{(i)})]}{(2\pi)|\Sigma|^{1/2}}, \tag{2.9}$$

$$\hat{\mathbf{z}}_{t,k}^{(i)} = \mathbf{H}\hat{\mathbf{x}}_{t,k}^{(i)} \text{ , and} \tag{2.10}$$

$$\Sigma \equiv \Sigma_{t,m,k}^{(i)} = \mathbf{H}\mathbf{P}_{t,k}^{(i)}\mathbf{H}^{\mathrm{T}} + \mathbf{R}_{m,k} \tag{2.11}$$

The superscript i in these equations is the ***PMHT*** iteration index. These weights are used to define the updated target measurement probabilities:

$$\pi_{t,k}^{(i)} = \frac{\bar{\omega}_{t,k}^{(i)} \pi_{t,k}^{(i-1)}}{\bar{\omega}_{t,k}^{(i)}}, \text{ where} \tag{2.12}$$

$$\bar{\omega}_{t,k}^{(i)} = \frac{1}{n_k} \sum_{m=1}^{n_k} \omega_{t,m,k}^{(i)}$$

where $n_k$ is the number **of** measurements in the scan at time k. The centroid measurement is now computed using:

$$\tilde{\mathbf{z}}_{t,k}^{(i)} = \frac{1}{n_k \bar{\omega}_{t,k}^{(i)}} \sum_{m=1}^{n_k} \omega_{t,m,k}^{(i)} \mathbf{z}(m,k), \text{ with covariance} \tag{2.13}$$

$$\tilde{\mathbf{R}}_{t,k}^{(i)} = \sum_{m=1}^{n_k} \left( \frac{\omega_{t,m,k}^{(i)}}{n_k \bar{\omega}_{t,k}^{(i)}} \right)^2 \mathbf{R}(m,k) \tag{2.14}$$

These quantities are now used in the standard Kalman smoothing algorithm to generate updated estimates for $\{\hat{\mathbf{x}}_{t,k}^{(i)}\}$ and $\{\mathbf{P}_{t,k}^{(i)}\}$. The algorithm now iterates until convergence is achieved (up to a maximum of 100 iterations in this research). Convergence is achieved when

$$\left( \sum_{k=1}^{T} \sum_{t=1}^{M} \left| \pi_{t,k}^{(i)} - \pi_{t,k}^{(i-1)} \right| \right) < 10^{-4} \tag{2.15}$$

This algorithm differs from the baseline described in Streit and Luginbuhl ([2], Section 5) in that each measurement has a distinct covariance matrix. Several variations of the above algorithm were also investigated involving the terms $\mathbf{R}_{m,k}$ in equation 2.11 and $\tilde{\mathbf{R}}_{t,k}^{(i)}$ in equation 2.14. The following alternative computations of $\tilde{\mathbf{R}}_{t,k}^{(i)}$ have been considered:

$$\breve{\mathbf{R}}_{t,k}^{(i)} = \sum_{m=1}^{n_k} \left( \frac{\omega_{t,m,k}^{(i)}}{n_k \bar{\omega}_{t,k}^{(i)}} \right) \mathbf{R}(m,k) \qquad (2.16)$$

$\hat{\mathbf{R}}_{t,k}^{(i)} = \mathbf{R}(m,k)$ associated with the measurement closest to the estimated position of track $t$ at time $k$. (2.17)

$\hat{\mathbf{R}}_{t,k}^{(i)} =$ the covariance computed using equations 2.3, where r and $\phi$ are computed from the estimated position of track $t$ at time $k$. (2.18)

Each of these in turn, including the original $\tilde{\mathbf{R}}_{t,k}^{(i)}$, have been used in equation 2.11 in place of $\mathbf{R}_{m,k}$. Of these methods, the estimated covariance defined in equation 2.18, $\hat{\mathbf{R}}_{t,k}^{(i)}$, has proved to be the most robust and has provided the overall best results. Hence, for most of the data runs discussed in section 3 below, the estimate $\hat{\mathbf{R}}_{t,k}^{(i)}$ has been used in both the Kalman smoothing equations and in equation 2.11 in place of $\mathbf{R}_{m,k}$.

The above algorithm includes a specific model for clutter based on the uniform density, and is specified via a clutter density parameter **p.** Clutter density parameters used in this research were in the range of $10^{-6}$ to 1 clutter return per square kilometer. The actual clutter densities used in the simulations tested here varied in the range of $3.3 \times 10^{-3}$ to $6.6 \times 10^{-2}$ clutter returns per square kilometer.

## 2.4 Initialization Algorithm

The initialization algorithm investigated in this research has been N-of-N, where N has varied from 2 to 5. Generally, M-of-N algorithms are used in a variety of realistic, fielded tracking systems and remain the most popular in general applications. The PMHT algorithm requires an initial state estimate for each track at each measurement time point in the entire trajectory, $\hat{\mathbf{X}}_t^{(0)}$, and initialization strategies based on a small number of measurement points at the beginning of a track can provide very poor estimates at the end of the track. Unfortunately, a tracking system built to operate in real time must operate under these kinds of constraints. We have concentrated on the N-of-N algorithm to establish some bounds on how many initial points are required for the algorithm to converge to a meaningful result.

The initialization algorithm used here can be divided into two distinct computational procedures: a gating strategy based on measurement and target speed uncertainty for 2-of-2 processing, and a least squares algorithm for N-of-N processing when $N > 2$. The 2-of-2 algorithm relies on some knowledge of target speed. This research has assumed targets of interest move with speeds between 2 and 10 knots. If the target velocity vector is known, and two successive measurements, $\mathbf{z}_k$ and $\mathbf{z}_{k+1}$, are obtained for the target, then the quantity

$$\chi^2 = (\mathbf{z}_{k+1} - \mathbf{z}_k)^T [\mathbf{R}_{k+1} + \mathbf{R}_k]^{-1} (\mathbf{z}_{k+1} - \mathbf{z}_k) \qquad (2.19)$$

has a noncentral chi-squared distribution, where $\{\mathbf{z}_k, \mathbf{R}_k\}$ are the measured track position and measurement covariance matrix at time k. It was found empirically that for target speeds between 2 and **10** knots and for tracking ranges used in the simulations described in Section **3** that a cutoff of $\chi^2 \leq 50$ kept all target associations. This produces an effective gate for 2-of-2 association of approximately 75 square kilometers.

The N-of-N algorithm relies on a least squares fit, assuming a constant velocity target. For example, in the 3-of-3 algorithm, the measurements at the first three measurement times are given by:

$$\mathbf{z}(t,1) = \mathbf{H}\mathbf{x}(t,1) + \varepsilon_{m,1}$$
$$\mathbf{z}(t,2) = \mathbf{H}\mathbf{x}(t,2) + \varepsilon_{m,2} = \mathbf{H}[\mathbf{A}(\Delta)\mathbf{x}(t,1) + \mathbf{w}(t,1)] + \varepsilon_{m,2}$$
$$\mathbf{z}(t,3) = \mathbf{H}\mathbf{x}(t,3) + \varepsilon_{m,3} = \mathbf{H}\big[\mathbf{A}(\Delta)[\mathbf{A}(\Delta)\mathbf{x}(t,1) + \mathbf{w}(t,1)] + \mathbf{w}(t,2)\big] + \varepsilon_{m,3}$$

For convenience, the process errors are set to zero $(\mathbf{w}(t,k) = \mathbf{0}$ for every $k)$, giving:

$$\begin{pmatrix} \mathbf{z}(t,1) \\ \mathbf{z}(t,2) \\ \mathbf{z}(t,2) \end{pmatrix}_{6x1} = \mathbf{F}\mathbf{x}(t,1) + \begin{pmatrix} \varepsilon_{m,1} \\ \varepsilon_{m,2} \\ \varepsilon_{m,3} \end{pmatrix}, \text{ where}$$

$$\mathbf{F} \equiv \begin{bmatrix} \mathbf{H} \\ \mathbf{H}\mathbf{A}(\Delta) \\ \mathbf{H}\mathbf{A}(\Delta)\mathbf{A}(\Delta) \end{bmatrix}_{6x4}$$

where the covariance associated with the error vector is

$$E\left\{\begin{pmatrix} \varepsilon_{m,1} \\ \varepsilon_{m,2} \\ \varepsilon_{m,3} \end{pmatrix}\begin{pmatrix} \varepsilon_{m,1} & \varepsilon_{m,2} & \varepsilon_{m,3} \end{pmatrix}\right\} = \begin{bmatrix} \mathbf{R}(m,1) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}(m,2) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}(m,3) \end{bmatrix} \equiv \Sigma_{\mathbf{R}}$$

Hence, the least squares estimate of $\mathbf{x}(m,1)$ is given by

$$\hat{\mathbf{x}}(t,1) = \left[\mathbf{F}^T\Sigma_{\mathbf{R}}^{-1}\mathbf{F}\right]^{-1}\mathbf{F}^T\Sigma_{\mathbf{R}}^{-1}\begin{pmatrix} \mathbf{z}(t,1) \\ \mathbf{z}(t,2) \\ \mathbf{z}(t,2) \end{pmatrix}, \text{ and} \qquad \textbf{(2.20)}$$
$$\hat{\mathbf{x}}(t,3) = \mathbf{A}(\Delta)\hat{\mathbf{x}}(t,2) = \mathbf{A}(\Delta)\hat{\mathbf{x}}(t,1)$$

Under the above assumptions, the quantity

$$\chi^2 = \left[\begin{pmatrix} \mathbf{z}(t,1) \\ \mathbf{z}(t,2) \\ \mathbf{z}(t,2) \end{pmatrix} - \begin{pmatrix} \hat{\mathbf{z}}(t,1) \\ \hat{\mathbf{z}}(t,2) \\ \hat{\mathbf{z}}(t,2) \end{pmatrix}\right]^T \Sigma_{\mathbf{R}}^{-1}\left[\begin{pmatrix} \mathbf{z}(t,1) \\ \mathbf{z}(t,2) \\ \mathbf{z}(t,2) \end{pmatrix} - \begin{pmatrix} \hat{\mathbf{z}}(t,1) \\ \hat{\mathbf{z}}(t,2) \\ \hat{\mathbf{z}}(t,2) \end{pmatrix}\right], \text{ where} \qquad \textbf{(2.21)}$$
$$\hat{\mathbf{z}}(t,k) = \mathbf{H}\hat{\mathbf{x}}(t,k)$$

has a chi-squared distribution with ***6*** degrees of freedom. The cutoff value for track formation at 3-of-3 was $\chi^2 < \mathbf{20}$, which gives an association probability for measurements derived from a real track of 0.9972. The 3-of-3 case also relies on a match using the 2-of-2 algorithm for the first and second measurement points and also for the second and third measurement points prior to implementing the least squares algorithm. Higher order associations project the 3-of-3 result into the future to determine likely measurements for association prior to obtaining the higher order least squares estimate.

vector is unrealistically large (> 10 knots here), it is scaled back to 10 knots to prevent the estimates obtained from projecting the initialized track forward in time from becoming unreasonably far from the measurements.

### *2.5 Probabilistic Data Association Filter*

The PMHT algorithm was compared with a probabilistic data association filter (PDAF) and a standard multi-hypothesis tracking (MHT) algorithm. The PDAF algorithm studied here is a standard algorithm as described in Chapter 3 of [4], with a modification to account for a different covariance matrix for each measurement. Also, a detection probability of unity is assumed. The processing sequence is as follows:

**1)** Track initialization using the N-of-N algorithm discussed above. All PDAF simulation results discussed below were run with N=3.

**2)** Gating. The gating procedure employs a chi-squared statistic computed using the current track state estimate and measurement as follows:

$$\chi^2 = \left(\mathbf{z}_{m,k} - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}\right)^T \left[\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^T + \mathbf{R}_{m,k}\right]^{-1} \left(\mathbf{z}_{m,k} - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}\right) \qquad \textbf{(2.22)}$$

A cutoff value was defined to give an association probability (assuming the measurement actually arises from the constant speed track) of 0.995. All measurements that gate with a track play a role in track updating.

**3)** Scoring the Association. For each measurement passing the above gate, a score is computed as follows:

$$\omega_{m,k} = \frac{\omega^*_{m,k}}{\sum_m \omega^*_{m,k}}, \text{ where} \qquad (2.23)$$

$$\omega^*_{m,k} = N(\mathbf{z}_{m,k}|\hat{\mathbf{z}}_k, \mathbf{P}_k, \mathbf{R}_{m,k})$$

Here, these quantities are as defined in section 2.2 above, with the denominator sum taken over all measurements that gate with the track. There are no iteration cycles for the PDAF, so there are no superscripts. Also, the dependence on the track model has been supressed because only one target track was studied using the PDAF.

**4)** Producing the updated state estimate. The updated state estimate is a linear combination of the standard Kalman update estimates for each measurement that gates with the track. The coefficients are the $\omega_{m,k}$:

$$\hat{\mathbf{x}}_{k|k} = \sum_m \omega_{m,k}\hat{\mathbf{x}}_{m,k|k} \qquad \textbf{(2.24)}$$

5) The covariance update is as follows:

$$\mathbf{P}_{k|k} = \left(\sum_m \omega_{m,k}\left(\mathbf{P}_{m,k|k} + \mathbf{K}\mathbf{G}_{m,k}\tilde{\mathbf{z}}_{m,k}\tilde{\mathbf{z}}^T_{m,k}\mathbf{K}\mathbf{G}^T_{m,k}\right)\right) - \tilde{\mathbf{x}}_{k|k}\tilde{\mathbf{x}}^T_{k|k}, \text{ where} \qquad (2.25)$$

$\mathbf{P}_{m,k|k}$ is the Standard Kalman Covariance Update for the $m^{th}$ Measurement,

$\mathbf{K}\mathbf{G}_{m,k}$ is the Associated Kalman Gain for the $m^{th}$ Measurement, and

$$\tilde{\mathbf{x}}_{k|k} \equiv \sum_m \omega_{m,k}\mathbf{K}\mathbf{G}_{m,k}\tilde{\mathbf{z}}_{m,k}$$

These equations, along with the standard Kalman prediction and update equations, complete the PDAF algorithm.

## 2.6 Multi-Hypothesis Tracking Algorithm

The MHT algorithm used here is based on Reid's measurement-oriented paradigm [5]. This algorithm has been widely discussed, and appears in several texts (e.g., Chapter 6 of [4] and Chapters 10 and 14 of [6]). As implemented here, target motion and measurement models are as discussed above (equations 2.1 and 2.2). The purpose of the MHT algorithm is to enumerate as many of the possible measurement-to-track association possibilities as is feasible, given finite processing time and computer memory.

A hypothesis consists of a collection of tracks together with an overall hypothesis score. To explain the algorithm and define the terms used here, consider the following example. Suppose that prior to the current scan time there is a single hypothesis in the system, consisting of a single track plus an additional measurement designated as clutter, along with the score for the hypothesis. Suppose further that the current scan consists of two measurements. Measurements are processed sequentially. The processing steps associated with the MHT are as follows:

1) The first measurement is compared with the predicted track position, and a chi-squared gate is used to exclude poor associations. The technique is the same as is used in gating for the PDAF.

2) If the measurement gates with the track, a new hypothesis is generated consisting of a new track (obtained by updating the previous track with the new measurement and assigning a new track number) and the original clutter measurement. All updates use a standard Kalman filter algorithm. The hypothesis score generated for the new hypothesis consists of the old hypothesis score multiplied by the score associated with this association. This score is computed as:

$$\frac{\mathrm{Exp}\left[-\frac{1}{2}\left(\mathbf{z}_{m,k}-\mathbf{H}\hat{\mathbf{x}}_{k|k-1}\right)^{T}\left[\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^{T}+\mathbf{R}_{m,k}\right]^{-1}\left(\mathbf{z}_{m,k}-\mathbf{H}\hat{\mathbf{x}}_{k|k-1}\right)\right]}{(2\pi)\left|\left[\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^{T}+\mathbf{R}_{m,k}\right]\right|^{1/2}} \qquad (2.26)$$

if the track in question has been constructed from more than one measurement point prior to being updated with this measurement. For a track that consists of only one measurement point, the multiplying factor is taken from the value of the emipirical distribution constructed from simulating 2-of-2 measurement associations on tracks moving with arbitrary headings at speeds between 2 and 10 knots (see section 2.4 above).

**3)** The measurement also generates a new hypothesis based on the new measurement being a new track. In this case the multiplying factor for the new hypothesis score is an adjustable parameter that represents the score for a new track. In practice this new track score factor is tuned for best results.

**4)** Finally, the measurement generates a new hypothesis based on the new measurement being an additional clutter point. The multiplying factor in this case **is** another parameter representing the score for clutter. As with the new track score factor, this clutter score parameter **is** tuned for best results.

***5)*** At this point either two or three new hypotheses have been generated from the original hypothesis, depending on whether or not the measurement gated with the track. More tracks in the original hypothesis would have generated more possibilities. The scores for the new hypotheses are then normalized so that the top scoring hypothesis has a score of 1, with the other hypotheses scaled accordingly. Should the number of hypotheses exceed the maximum allowable number, lower scoring hypotheses are discarded and tracks that only appear in these discarded hypotheses are erased.

**6)** Now the second measurement in the scan can be processed, and the same procedure is followed for each of the existing hypotheses, with the following modification: the measurement may not gate with a track that has been created or updated during earlier processing in this scan (precluding the possibility of multiple measurements from the same target in a single scan).

The above outline specifies the overall outline of MHT processing. This MHT algorithm has no cluster processing. In addition, the algorithm prevents tracks that have not received a

measurement association for three scan cycles from being updated, although such tracks are not erased from the track list.

## 2.7 Attribute Data Processing

This year's research effort concluded with a study on the use of attribute data in the PMHT algorithm. The attribute study assumed that a measure of target amplitude was available to the processing algorithm in addition to the range-bearing measurement. The measured amplitude was drawn from a Rayleigh distribution with specified mean. The Rayleigh distribution has the form:

$$f(a) = \left(\frac{a}{\sigma^2}\right) Exp\left[-\frac{a^2}{2\sigma^2}\right], \quad a > 0 \qquad (2.26)$$

with mean value given by $a\sqrt{\pi/2}$. Here, target and clutter measurements were assigned different means. The clutter model used $a = 1$, while the target model used $a > 1$, where several values were tested. The inclusion of attribute data in the PMHT consisted of adjusting the weights found in equations 2.6 and 2.7:

$$\omega_{t,m,k}^{*(i+1)} = f(a_{m,k}) N(z_{m,k} | \hat{z}_{t,k}^{(i)}, P_{t,k}^{(i)}, R_{m,k}) \text{ for a target track,} \qquad (2.27)$$

$$\omega_{t,m,k}^{*(i+1)} = \rho f(a_{m,k}) \text{ for clutter,} \qquad (2.28)$$

where $a_{m,k}$ is the attribute portion of measurement m at time k. In this year's research, attribute data was only studied in conjunction with the PMHT. Generally, attribute data in our simulation results had only marginal effect for target values below $a = \sqrt{10}$, representing an order of magnitude difference in the power level between targets and clutter.

# 3. Simulation and Results

All of our comparison studies were performed using simulated data. An example simulation run is presented in Figure 1. Here, actual target motion is along the straight line moving from upper left to lower right in the figure. Thirty target range-bearing measurements are generated using additive, Gaussian noise, where the range standard deviation is 100 meters and the bearing standard deviation is 3 degrees. The sensor is located at (0,0), so the target is moving predominantly in the cross-range dimension at ranges beyond 30 kilometers with respect to the sensor in the scenario depicted in this figure. Target measurements are pictured in the figure using "*" symbols, while clutter appears as circles. Scans of data are generated every 4 minutes for two hours. Actual target motion is 5 knots in all scenarios. For the scenario depicted here, there are 5 clutter measurements and 1 target measurement generated at each measurement scan. This corresponds to a clutter density of $1.67 \times 10^{-2}$ clutter points per square kilometer. Target probability of detection was assumed to be 1 throughout all testing detailed in this report. Clutter densities were varied fiom a low of $3.33 \times 10^{-3}$ to a high of $6.67 \times 10^{-2}$ clutter points per square kilometer in these studies.

## 3.1 PMHT Implementation and Results

Algorithm testing began using three target motion geometries and low clutter. These geometries were: a single track moving in a straight line with constant velocity, two crossing tracks moving in straight lines with constant velocity, and a track that performs a turn half way through the scenario, generating two straight line segments. Algorithm testing and adjustments continued until reasonable convergence results were obtained on the single non-maneuvering track

scenario and the crossing track scenario at clutter densities of $1.67 \times 10^{-2}$ clutter points per square kilometer (as depicted in Figure 1).

The PMHT algorithm modifications used to obtain these convergence results consist of the following:

1) Equation 2.18 is used for the measurement covariance matrix. Other implementations of the debiased conversion covariance matrices (i.e., the use of equations 2.14, 2.16 or 2.17) produced inferior or unsatisfactory results.
2) N-of-N Initialization was used, where N=5. Values of N=2, 3, and 4 were also tried, but results were not satisfactory.
3) Measurements beyond the Chi-squared cutoff value of 0.995 from the estimated track position at each scan time have their scores set to a low constant value of $10^{-20}$, rather than using the actual score generated by equation 2.9.
4) PMHT processing proceeds by batch processing an additional increment of five scans of data at a time. The initial track estimates for the incoming set of five data scans is obtained by predicting the results of the previous converged track estimates into the future. Then the algorithm runs on all data beginning at time zero up through the new data, iterating until convergence, equation 2.15, is obtained (or the maximum number of 100 iterations is exceeded). This process continues until all data scans have been processed.
5) A 10 knot maximum speed is imposed when predicting the initial track estimates for the next 5 data scans. The actual target estimates generated at algorithm convergence have no speed limitations.
6) For the straight tracks, the plant noise parameter q (section 2.1) is set to 1. For the maneuvering track q was set to values up to 1000, but the algorithm was still ineffective (as was the PDAF).

Figure 2 shows a typical convergence result obtained for the PMHT algorithm on a single straight-line track. Here, circles are actual target positions at each measurement time, and "*" symbols are the converged smooth estimates of target position. Clutter density for this scenario is $1.67 \times 10^{-2}$ clutter points per square kilometer. Note that the true target trajectory is aligned well with the estimates, but the converged estimates are at a slower speed than the actual target.

Figure 3 shows a typical convergence result for the PMHT algorithm on crossing targets. Here, the crossing track from lower left to upper right is closely aligned with a bearing line from the sensor, while the other track from upper left to lower right is along the cross-range dimension as before. Clutter density is again $1.67 \times 10^{-2}$ clutter points per square kilometer. Some alignment and velocity discrepancies are evident from the figure.

Figure 4 shows a typical convergence result for the PMHT algorithm on the maneuvering track. Actual track positions (circles) proceed from upper left to center right in the cross-range dimension, when the maneuver occurs. The target then proceeds down to the lower left. The converged estimate ("*" symbols) clearly loses the target at the turn, and special processing will be required to track through a sharp maneuver.

In order to quantify our results, mean distance errors were computed for the final converged target position estimates at each scan time. Means were taken over 100 simulation runs. The crossing track scenario was investigated first, and the mean position error results for this scenario are reported in Figure 5. This figure shows 4 curves. The lowest curve is the result of running the Kalman smoother using the actual target measurements (i.e., no clutter) and taking

mean distance errors, where the mean is computed over 1000 simulations. Hence, this curve represents a theoretical minimum for algorithm performance in the absence of clutter. The highest and most irregular curve is generated by taking the mean distance between the target measurements and true target positions, again with means generated over 1000 simulations. Hence, this curve represents the performance of the unfiltered measurements in the absence of clutter. These two curves represent operating bounds for the actual algorithm, which cannot achieve the lower curve due to the interference of clutter and is not very useful as a target locator if it strays too far above the upper curve. The two middle curves represent mean distance errors generated over 100 simulation runs by the PMHT algorithm. The lower of these two curves (marked by "*" symbols) corresponds to errors with respect to the bearing line target (from lower left to upper right in figure **3).** The upper of the two middle curves (marked by "X" symbols) is for the cross-range target. Hence, the cross-range target is the more difficult target in this scenario, although both sets of distance errors are comfortably inside the raw measurement error.

The above results were obtained at a clutter density of $1.67 \times 10^{-2}$ clutter points per square kilometer. If the clutter density is doubled, results are much worse, as shown by the distance error curves depicted in Figure 6. Again, mean distance errors have been computed over 100 simulation runs, and again the cross-range target generates much poorer position estimates. Hence, the cross-range target motion appears to be the most affected by clutter. Therefore most of our effort has been focused on the cross-range target geometry.

Figure 7 depicts actual target positions and measured positions for this geometry, where measurements are connected via line segments to illustrate the relative volatility being generated by the **3** degrees of measurement error at these ranges. Study of outliers (scenarios where final convergence results were above the measurement error line at the final measurement time) indicated that while the 5-of-5 initialization procedure tended to prevent track estimates from going off at large angles to the actual target motion, final velocity estimates were often too fast, even with the 10 knot limit. Figure **8** depicts a typical outlier convergence result. Hence, we adjusted our 10 knot limit down to 2 knots. This means that the initial target speed estimate used to initialize the next five data points for the PMHT at the beginning of processing each new batch of 5 scans of data was set to a maximum of 2 knots for predicting the initial estimates (item # ***5*** in our list of PMHT modifications given above). The final speed could, however, settle out to any value. The results of this change, along with subsequent modifications to the PMHT, are reported in the comparative performance section (section **3.3)** below.

### 3.2 MHT Implementation

The MHT processing algorithm is discussed in section ***2.6*** above. Here, specific aspects of the algorithm that affect the way it is to be compared with other algorithms are discussed. First, in comparing distances between actual target positions and estimated positions, there is the issue of which track in the MHT track list should be used for the comparison. We have chosen two tracks from the track list as indicative of MHT tracking performance. The first track is the "established" track in the current top scoring hypothesis that is closest to the current target position. By "established" we mean a track that includes at least N measurement points to prevent short, spurious tracks from giving a false sense of this error. In this research we have chosen N=3 to match the start-up criterion for the PDAF algorithm we have implemented. This track will be referred to as the top hypothesis track.

A second track from the MHT track list that is logical to use for comparison is the track containing the most true target points. In our simulations, measurements generated fiom target positions are numbered beginning from 1 and going as high as 60, for scenarios that feature two targets. Measurements generated as clutter, however, are numbered beginning from 101 and going as high as required. Hence, although none of our processing algorithms use this information in creating or updating tracks, it is possible to know how many actual target points appear in any track by noting the measurement numbers. This track, the one with the most actual target measurements, is referred to as the best track, even though it may not appear in the top scoring hypothesis.

A sample track list generated fiom a scenario run consisting of a single, non-maneuvering target track is depicted in Table 1. This table was the final table generated during the scenario, so it represents a snapshot at the end of processing all measurement scans. Also present in the track table, but suppressed in these displays, is the current state and covariance estimates for each track in the table. In this table, track #413 is the best track. It contains 30 total measurements (column 2), and all 30 of those measurements are fiom the actual target (column 3). The first three measurements in this track have ID numbers 1, 2, and 3 (columns 4, 5, and 6), indicating that it originated from the first three target measurements. The last update time for this track was 116 seconds into the scenario (column 7), which was the last measurement scan time in the scenario.

**Table 1: Exemplar Track Table Generated by the MHT Algorithm for a Single Track Scenario with No Maneuvers**

| Track ID | # Points in Track | # Tgt Pts in Track | First Meas. ID in Track | 2nd Meas. ID in Track | 3rd Meas. ID in Track | Time of last Update (s) |
|---|---|---|---|---|---|---|
| 2 | 1 | 0 | 101 | 0 | 0 | 0 |
| 3 | 1 | 0 | 102 | 0 | 0 | 0 |
| 4 | 1 | 0 | 103 | 0 | 0 | 0 |
| 5 | 1 | 0 | 104 | 0 | 0 | 0 |
| 6 | 1 | 0 | 105 | 0 | 0 | 0 |
| 413 | 30 | 30 | 1 | 2 | 3 | 116 |
| 414 | 30 | 29 | 1 | 2 | 3 | 116 |
| 415 | 30 | 29 | 1 | 2 | 3 | 116 |
| 416 | 30 | 29 | 1 | 2 | 3 | 116 |
| 417 | 30 | 28 | 1 | 2 | 3 | 116 |
| 418 | 30 | 29 | 1 | 2 | 3 | 116 |
| 419 | 30 | 28 | 1 | 2 | 3 | 116 |
| 420 | 30 | 28 | 1 | 2 | 3 | 116 |

Table 1 illustrates some of the shortcomings of our current MHT processing. First, there is no track delete unless all hypotheses containing a given track are themselves deleted. Hence, tracks represented in the first five rows of Table 1 remain in the table even though they have not been updated since they were formed during the processing of the first measurement scan. Old tracks are prevented fiom updating with new measurements once they have failed to update during two consecutive measurement scan cycles, but they are not deleted. This feature has some usefulness in a research environment, as will become evident in our discussion of Table 2 below, but it would not be present in a fielded system.

The second shortcoming is our lack of a track merge feature. The last eight tracks depicted in Table 1 are all essentially the same track, with one or two minor measurement substitutions. These tracks are all active throughout the entire scenario, and they could potentially clog both the hypothesis list and the track list, detracting from both algorithm efficiency and tracking performance. Here we have the luxury of carrying a large number of hypotheses because we are not operating in real time, and keeping all such tracks gives us an accurate count of the maximum number of actual target measurements that can be associated together by the MHT algorithm at various clutter levels. However, a track merging algorithm would be a requirement for any fielded system.

Figure 9 shows a graph of the minimum, maximun and average number of target points in the best track, where these statistics are computed over 100 simulation runs. The clutter was at a moderate density of $1.67 \times 10^{-2}$ clutter points per square kilometer. Since there are a maximum of **30** target measurement points, the upper curve in the figure demonstrates that in at least 1 of the 100 simulation runs all target points were associated into one track. The middle curve indicates that the average number of target points in the best track was 24. The bottom curve indicates that the minimum number of actual target measurements associated into the best track at the end of the scenario was 11 target points.

Figure 10 shows a graph of the mean distance error between the actual target location and the MHT track estimate for both the top hypothesis track ("o" symbols) and the best track ("*" symbols). Mean errors are taken over 100 simulation runs with the clutter density given above. Here the best track tends to give better results, but they are both virtually identical at the end of the scenario.

**Table 2: Exemplar Track Table Generated by the MHT Algorithm for a Single Track Scenario with a Turn**

| Track ID | # Points in Track | # Tgt Pts in Track | First Meas. ID in Track | 2nd Meas. ID in Track | 3rd Meas. ID in Track | Time of last Update (s) |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 2 | 1 | 0 | 101 | 0 | 0 | 0 |
| 3 | 1 | 0 | 102 | 0 | 0 | 0 |
| 4 | 1 | 0 | 103 | 0 | 0 | 0 |
| 5 | 1 | 0 | 104 | 0 | 0 | 0 |
| 6 | 1 | 0 | 105 | 0 | 0 | 0 |
| 24 | 3 | 3 | 1 | 2 | 3 | 8 |
| 25 | 3 | 2 | 105 | 2 | 3 | 8 |
| 28 | 3 | 1 | 105 | 108 | 3 | 8 |
| 118 | 5 | 5 | 4 | 5 | 6 | 28 |
| 121 | 5 | 2 | 4 | 5 | 126 | 28 |
| 213 | 10 | 6 | 9 | 10 | 155 | 68 |
| 329 | 12 | 12 | 19 | 20 | 21 | 116 |
| 330 | 12 | 11 | 19 | 20 | 21 | 116 |

We also explored the maneuvering track scenario using the MHT. In this case the target maneuvers, as shown in figure 4 above. An exemplar track table is presented in Table 2, where the track table is taken at the end of the scenario. Here, the first 12 tracks have ceased to update, and only the last 2 tracks are still viable at the end of the scenario, and these two are essentially

the same track. Track 24 initiated from the first 3 target measurements (columns 4-6), but failed to update thereafter. Track 213 initiated with track measurement 9 and updated 9 more times before it died out. Six of its measurements are actual target measurements (column 3), making it the longest surviving track prior to the maneuver, which occurred with target measurement 16, 60 seconds into the scenario. Hence track 213 is the best track shown in the table prior to the maneuver. After the maneuver the target was reacquired by tracks 329 and 330 at target measurement 19, 72 seconds into the scenario. These tracks remained locked onto the target until the end of the scenario. Track 329 ended up with 12 target measurements, making it the best track pictured in this simulation run.

Figure 11 gives the average number of target points in the best track for this maneuvering track scenario. This graph shows that a maximum of 18 target points appear in any of the best tracks in the 100 simulation runs. Such a track would be initiated with the first target point and continue through point 16, when the maneuver occurs. Two additional points would be added before the maneuver takes subsequent measurements out of reasonable gating range. Thereafter, the target would have to be reacquired by starting another track. The average number of target points in the best track for this scenario is around 14, and the shape of the middle curve in the figure indicates that these tracks are produced by tracks initiated early in the scenario, not during or after the turn.

Figure 12 gives the average number of target measurement points in the best track for tracks initialized both before and after the turn. The upper curve is the same as the middle curve in figure 11 above, and gives the mean number of target measurements in tracks initialized prior to the maneuver. The lower curve shows that the average number of target points appearing in the best track initialized at the 56 second mark in the scenario or later is 11 by the end of the scenario. Hence, reacquisition is not as fast as was original acquisition for the target track, but reacquisition does occur.

Figure 13 shows the mean distance errors using the MHT for the maneuvering track scenario. Again, both the top hypothesis track ("o" symbols) and the best track ("*" symbols) are shown. Mean errors are taken over 100 simulation runs with the clutter density given above. Here the best track performs best prior to the maneuver, but the top hypothesis track performs best following the maneuver.

The actual setup parameters used for the MHT in this research were chosen to permit the reacquisition of the target following the maneuver in the maneuvering target scenario. These same parameters were then used to produce the results given above and the results given in the comparative performance section below.

### 3.3 Comparative Algorithm Performance

The algorithms were compared directly using the equations and techniques described in the sections above, with mean distance errors as the performance measure. All means were computed over 100 simulation runs. Figure 14 shows the comparative distance errors for the various algorithms on a single straight track scenario, where the clutter density is $1.67 \times 10^{-2}$ clutter points per square kilometer. Among these algorithms, only the PMHT uses a smoothed estimate, and its performance should be superior at all except the last data point in the scenario because at other points the PMHT is using more data than the other algorithms. In this figure the lowest curve at the right hand side of the graph is the smoothed estimate, taken over 1000 simulation runs, using actual target measurements only. The highest curve is the mean measurement error using target measurements only, again taken over 1000 runs. These curve

duplicate the curves given in figures 5 and 6, section 3.1 above. For this straight-line track scenario at this clutter density, the PMHT algorithm ("*" symbol) performed as well as the two track types generated by the MHT ("o" for best track and "+" for top hypothesis track). The PDAF ("x" symbol) and the nearest neighbor tracker (dashed line), were poorer performers overall, although all algorithms performed better than the raw target measurements themselves, even though the algorithms had to deal with clutter.

Figure 15 demonstrates the effect of doubling the clutter density. Here; the nearest neighbor tracker has been dropped. Note that the PMHT is now out-performed by both tracks associated with the MHT, but it still performs better than the PDAF.

Figure 16 shows the effect of these algorithms on the turning track scenario. Here it is clear that only the MHT has the capability to reacquire the track after the maneuver. The results given in figures 14, 15, and 16 were reported at the Asilomar Conference in November of this year.

### 3.3.1 Algorithm Shortcomings and Improvements

Unfortunately, the modified PMHT algorithm used to obtain the above comparative results proved unacceptable when used on the crossing track scenario. In this instance, the track moving along the bearing line of the sensor, which had proved to be the easier of the two tracks to estimate (as reported in figures 5 and 6, section 3.1 above), was no longer giving reasonable convergence results. The algorithm had been changed to initialize future estimates at a speed of 2 knots (section 3.1 above), and this low limit was causing havoc with the convergence of the bearing line target track, whether it was being measured in the crossing track scenario or by itself. This led us to revisit some fundamental aspects of the algorithm and our modifications to it.

We identified three areas where improvements might be made. The first of these was the use of the debiased conversion measurements and the corresponding problems we had earlier with the covariance estimate. Here, we decided to explore the Extended Kalman Smoother (EKS) algorithm in lieu of the debiased conversion. The second of these was a revisit of the earlier concept of covariance deflation. The third was to examine the method we were using to limit the speed for initializing the PMHT.

The results of the EKS study is summarized in Figure 17. The curves in this figure represent means of 500 simulation runs on the cross-bearing track scenario, where the clutter density is $3.3 \times 10^{-2}$ clutter points per square kilometer. Here, the solid line on the graph represents the debiased conversion algorithm that we have been running up to this time. The tolerance for convergence for the debiased conversion measurements was $10^{-4}$ (equation 2.15), as has been the case throughout our earlier runs. These 500 debiased conversion simulation runs took 738 minutes to complete, and there were 74 cases where the algorithm did not converge to a final estimate that was less than the mean measurement error. The dashed line on the graph represents the EKS algorithm results on 500 simulation runs at the same tolerance for convergence, $10^{-4}$. This EKS algorithm took only 242 minutes to complete, and there were only 68 cases where the algorithm did not converge to a final estimate that was less than the mean measurement error. Because the EKS ran so much more quickly than the debiased conversion algorithm, we tested this algorithm for convergence at the more stringent tolerance of $10^{-8}$. This is reported in the dash-dot curve in figure 17. Here, the EKS algorithm completed the 500 simulation runs in 240 minutes with only 57 cases where the final estimate exceeded the mean measurement error. A further refinement of the tolerance to $10^{-10}$ did not improve EKS

performance. It is clear from these curves and statistics that the EKS at the more exacting tolerance of $10^{-8}$ gives comparable performance at a much faster convergence rate. Hence, we have modified our processing to use the EKS at a tolerance of $10^{-8}$.

Further testing of the covariance deflation strategy was not productive. As implemented here, the measurement covariance was initially inflated by a factor of **3** for the first iteration of the algorithm, then by a factor of 2 for the second iteration, and finally by a factor of 1 for the third and all succeeding iterations. Figure 18 shows the comparative results between this covariance deflation algorithm and the standard PMHT. Here, the line with circles is the standard algorithm and the dashed line is the covariance deflation algorithm, mean values being taken over 100 simulation runs. Other variations of this technique were similarly unimpressive.

Finally, we reexamined our method for limiting the speed of the target for algorithm initiation. It was fundamentally unaesthetic to limit the target to 2 knots when the estimated speeds of interest are between 2 and 10 knots, even when the algorithm is free to subsequently raise the speed. Hence, we decided to find a way to use the more justifiable 10 knot limit. We selected a baseline state estimate and measurement time to use in beginning our initialization for the next batch of **5** measurement scans. This baseline was chosen to be the most current scan time and the most current state estimate. If this state estimate had a speed in excess of 10 knots, then the velocity of the baseline state estimate was reduced to reflect a speed of 10 knots. Then state estimates at both past and future times were generated from the adjusted baseline state with its new velocity vector. These estimates were then used to initialize the algorithm for the next round of batch processing. Hence, the algorithm is still permitted to produce a final track estimate with any speed, but the initial estimate is speed bounded at every initial state vector. This algorithm has proved to work quite well.

Figures 19 and 20 repeat figures 14 and 15 with the EKS processing and speed initialization procedures discussed above used to modify the PMHT algorithm. (In these two figures the mean results for the revised PMHT are taken over 500 simulation runs. Other algorithm results are taken over 100 simulation runs, as before.) Now we note that the revised PMHT is slightly better than the MHT at low clutter densities, but still not as good as the clutter density increases. Figure 21 repeats Figure 6 with this modified PMHT acting on crossing tracks, yielding much better results.

### 3.4 Attribute Processing Results

Attribute information processing was explored using the equations detailed in section 2.7 above. We first set out to determine what ratio of target/clutter amplitudes would enhance the baseline PMHT results. This was done by fixing the parameter a for clutter in the Rayleigh distribution, equation ***2.26,*** at 1 and varying the parameter for target measurements. Values explored for targets were: $1, \sqrt{2}, 2, \sqrt{10}$, and $\sqrt{20}$, corresponding to power ratios of 0dB, 3dB, 6dB, 10dB, and 13dB. Figure 22 gives mean estimation errors using attribute processing. Mean values are taken over 100 simulation runs, where the clutter density is $3.3 \times 10^{-2}$ clutter points per square kilometer. Here, the dash-dot line is the standard PMHT without attribute data. The curves indicated by "o" (0dB), "+" (3dB), and "x" (6dB) actually show a degradation in performance, while the curves indicated by "*" (10dB) and dashes (13dB) indicate performance improvement. Hence, it seems clear that a minimum power ratio of 10dB is required for reasonable results.

Figure 23 shows the effect of varying the initialization constant, N in N-of-N, using the new attribute data, where a 10dB power ratio is used throughout. Again, the dash-dot curve is

the standard PMHT with no attribute data and N=5. The curve designated by circles is with N = 3, the "+" curve is N=4, and the "*" curve is N=5. The clutter density is $1.67 \times 10^{-2}$ clutter points per square kilometer. Here, the attribute data gives similar results at N=3 as is obtained without any attribute processing and N=5.

Figure 24 again shows the effect of varying the initialization constant, where the clutter density has been doubled over the level used in figure 23. Here we find that attribute processing (10dB) with N=3 (circles) is not as good as no attribute processing and N=5 (dash-dot curve). However, attribute processing (10dB) and N=5 shows superior results. Therefore, at higher clutter levels it is not possible to reduce N and make up the difference with 10dB of attribute information, which indicates that the PMHT algorithm is quite sensitive to its initialization routine.

## 4. Conclusions

This research has demonstrated that the PMHT algorithm is a viable candidate in the data association and tracking arena. It has proven to be superior to the PDAF in the scenarios examined here, and has outperformed the MHT in low clutter, although it is not as good as the MHT in higher clutter. The baseline PMHT algorithm will not track through a target maneuver, but most algorithms designed to operate in clutter require special processing to track a target through a turn. (Usually this entails reacquiring the target on its new course and linking the two tracks.) Hence, this is not a glaring weakness. Furthermore, the algorithm is easily modified to accept attribute data, although a 10dB signal-to-clutter power ratio is rather high.

Algorithm initialization remains an area of concern, as N=5 is a rather stringent criterion, especially when scenarios with Pd < 1 are included. Unfortunately, the algorithm seems to be quite sensitive to changes in initialization when this initialization must be based on time-ordered measurements, as is the normal processing case for a real-time, fielded system.

Clearly, more work needs to be done on initialization, on additional processing to track through turns, and on developing additional strategies for employing attribute data to increase its impact on algorithm performance.

## 5. References


[1] Don Lerro and Yaakov Bar-Shalom, "Tracking with Debiased Consistent Converted Measurements Versus EKF", *IEEE Transactions on Aerospace and Electronic Systems,* Vol. 29, No. 3, July 1993.

[2] Streit, Roy L., and Luginbuhl, Tod E., Probabilistic Multi-Hypothesis Tracking , NUWC-NPT Technical Report 10,428, 15 February 1995.

[3] R. L. Streit and T. E. Luginbuhl, "Maximum Likelihood Method for Probabilistic Multi-Hypothesis Tracking," Proceedings of the SPIE International Symposium on Signal and Data Processing of Small Targets, SPIE Proceedings Vol. 2335-24, pp. 394-405, Orlando, FL, 5-7 April 1994.

[4] Bar-Shalom, Yaakov, and Xiao-Rong Li, *Multitarget-Multisensor Tracking: Principles and Techniques,* 1995, Published by Yaakov Bar-Shalom and obtainable from him at (203) 486-4823 or by contacting him at ybs@ee.uconn.edu.

*[5]* D. B. Reid, *"An* Algorithm for Tracking Multiple Targets," ***IEEE Transactions on Automatic Control,* AC-24,** pp. 843-854, December 1979.

[6] Blackman, **Samuel S.,** Multiple-Target Tracking with Radar Applications, Artech House, Norwood, MA, 1986

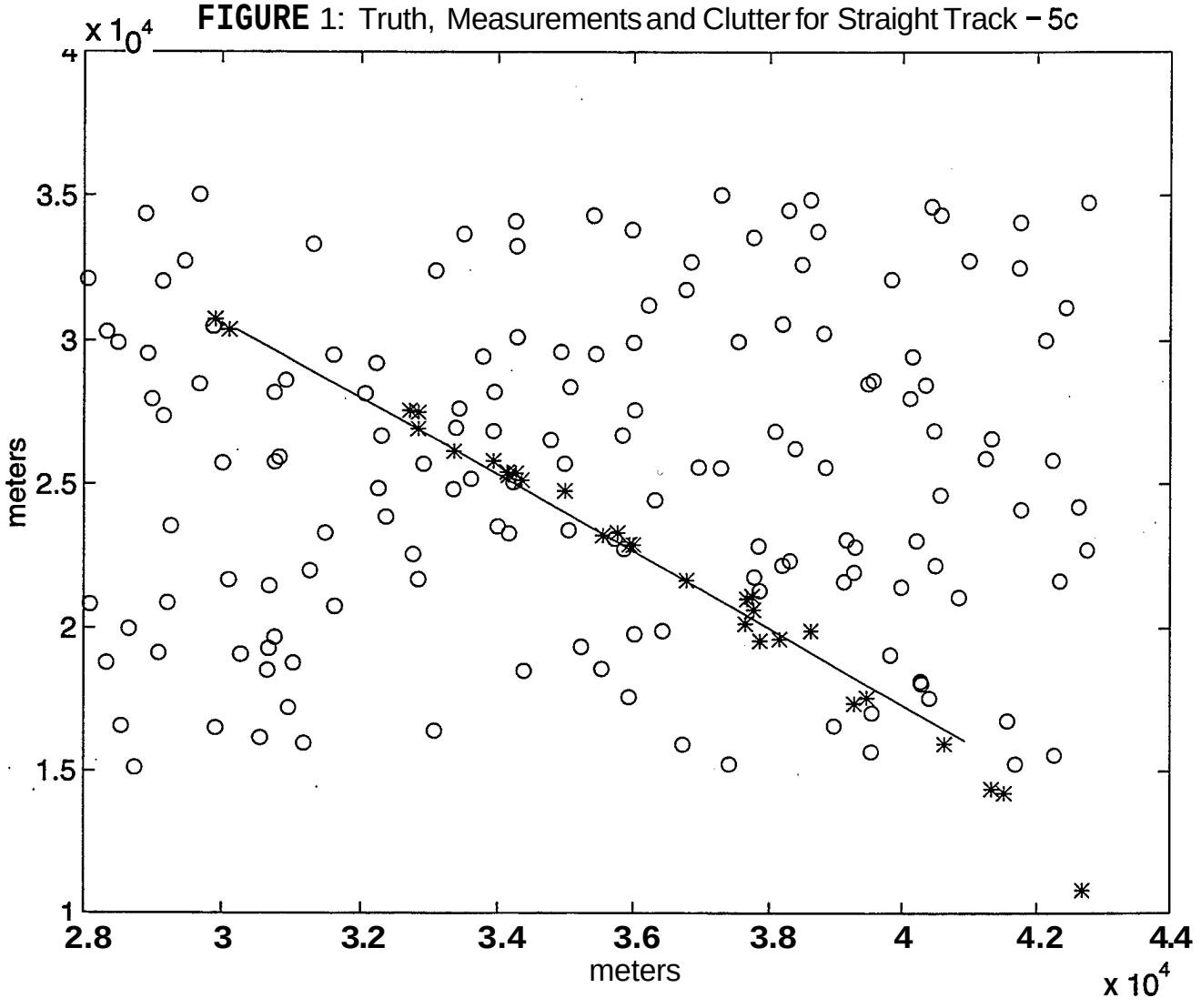



**FIGURE** 1: Truth, Measurements and Clutter for Straight Track – 5c

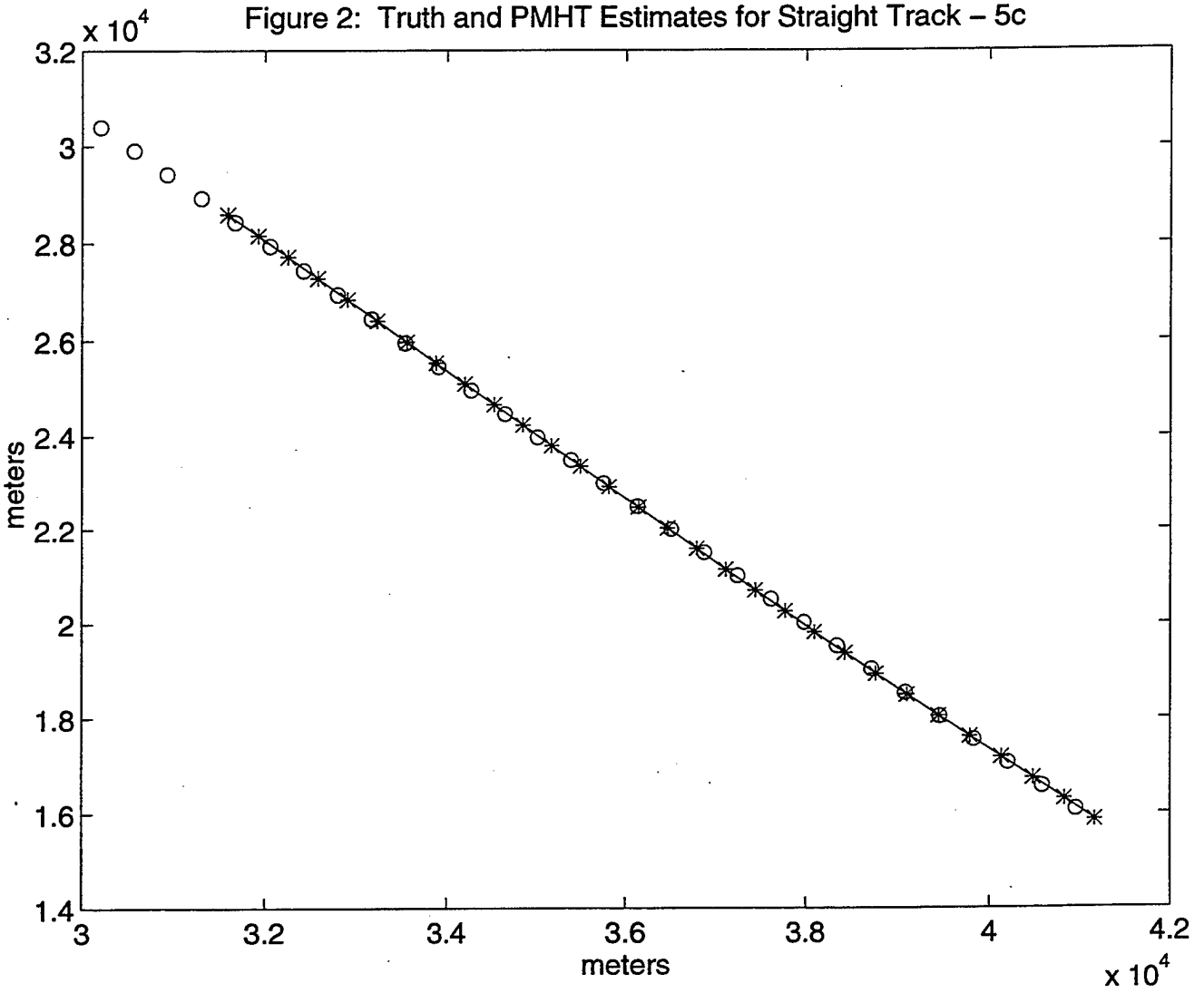



Figure 2: Truth and PMHT Estimates for Straight Track – 5c

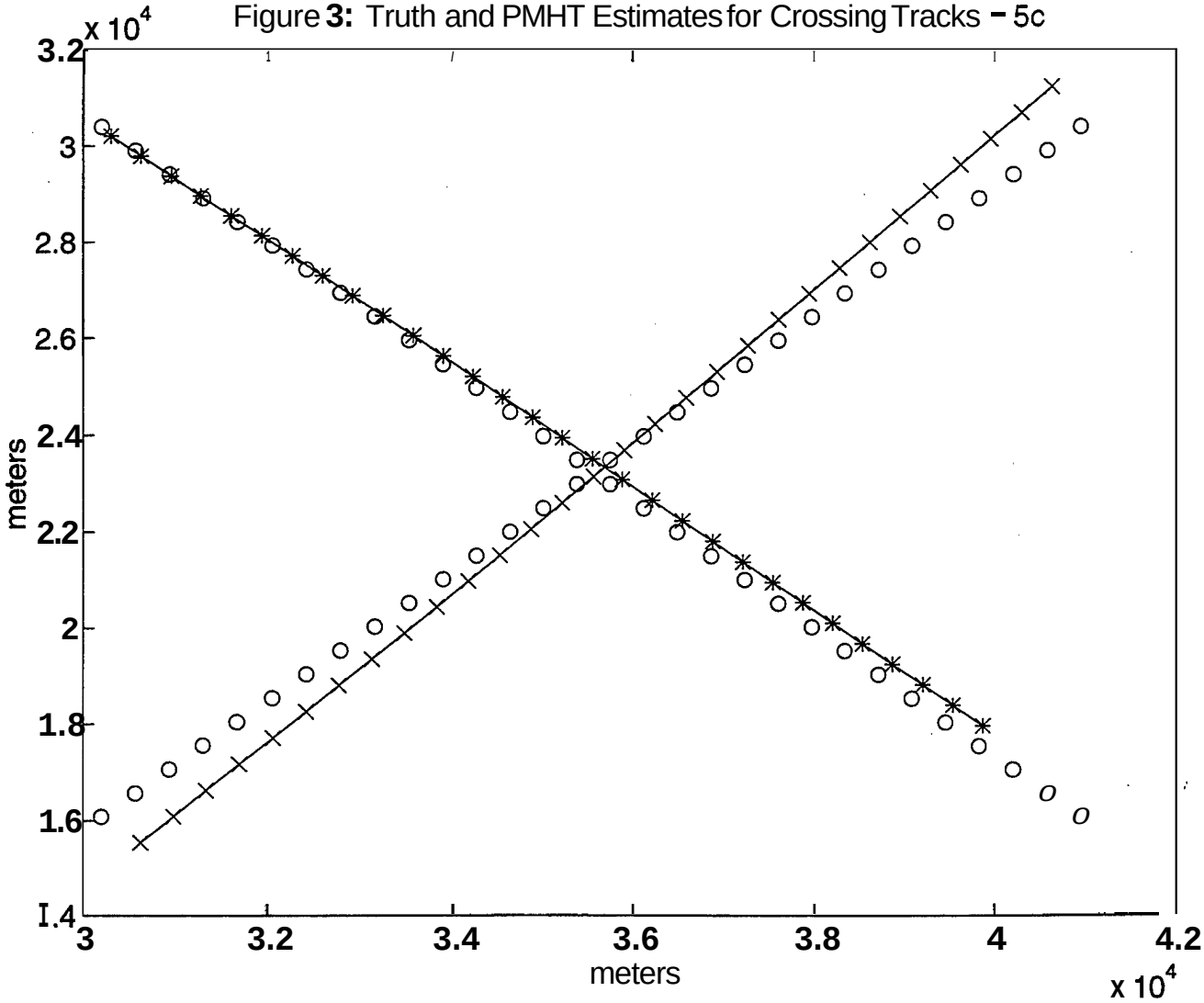
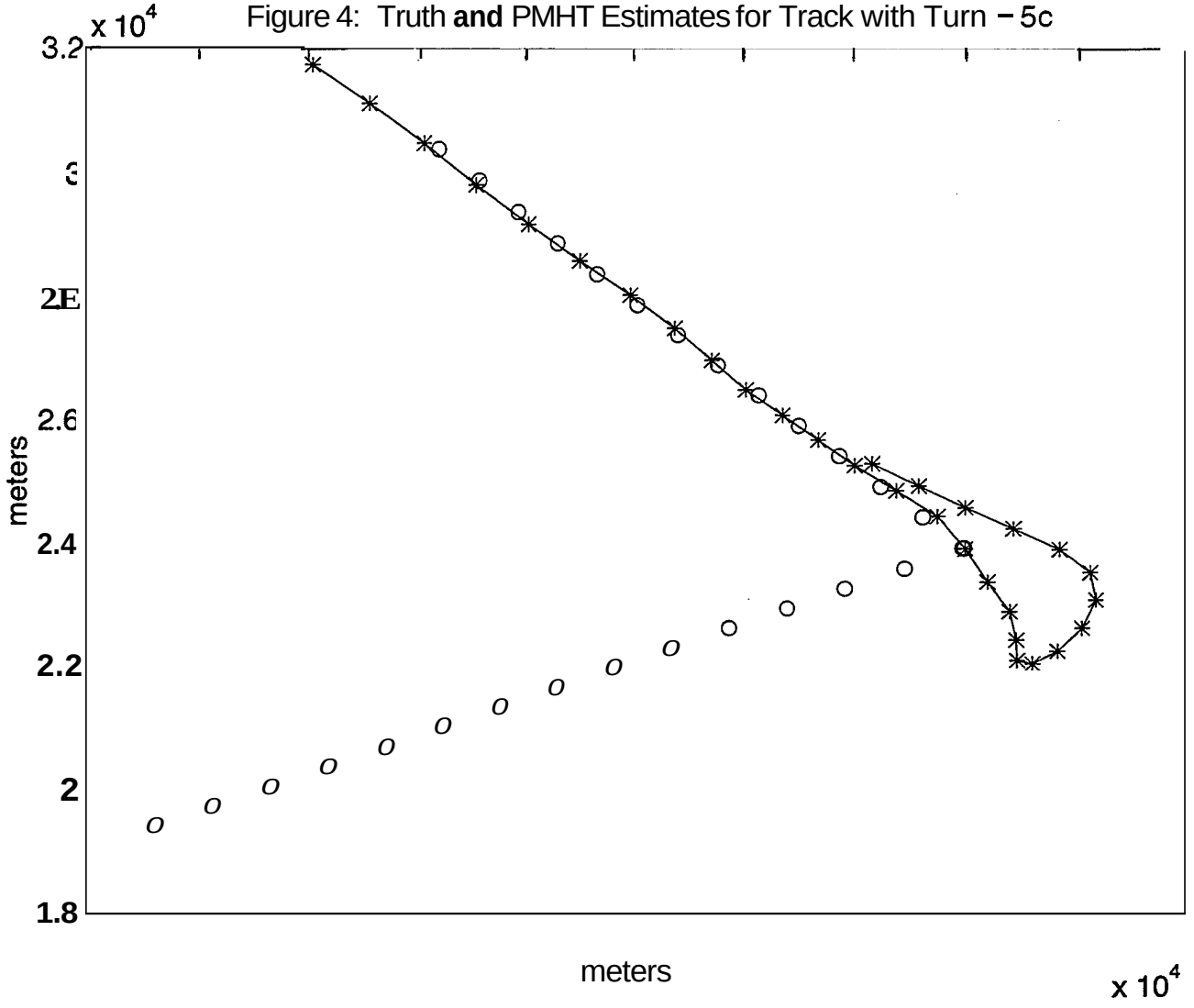



Figure 3: Truth and PMHT Estimates for Crossing Tracks – 5c

Figure 4: Truth **and** PMHT Estimates for Track with Turn – 5c



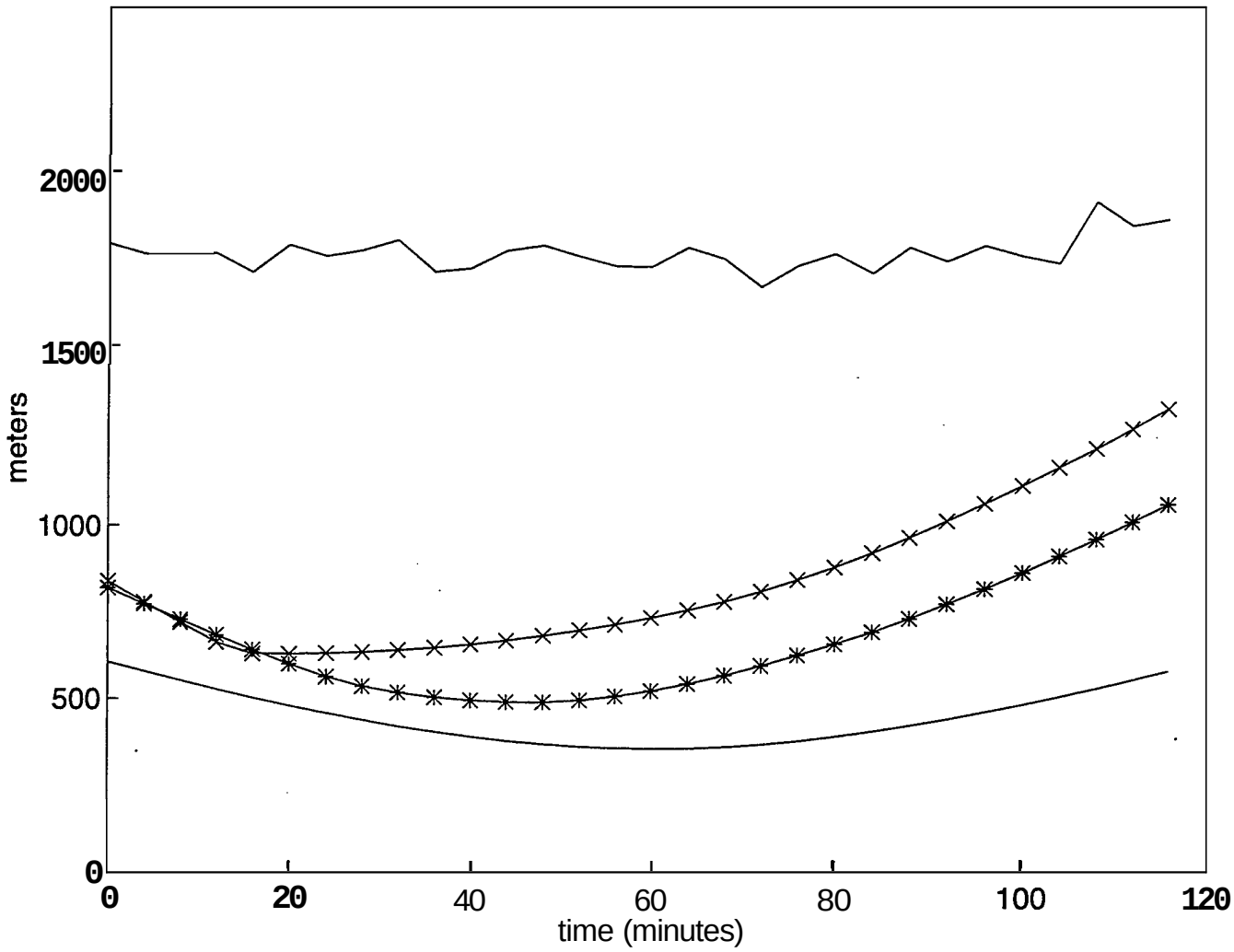

2000
1500
1000
500
0
meters
0
20
40
60
80
100
120
time (minutes)

**Figure 6: Mean PMHT Distance Errors for Crossing Tracks – 10c**

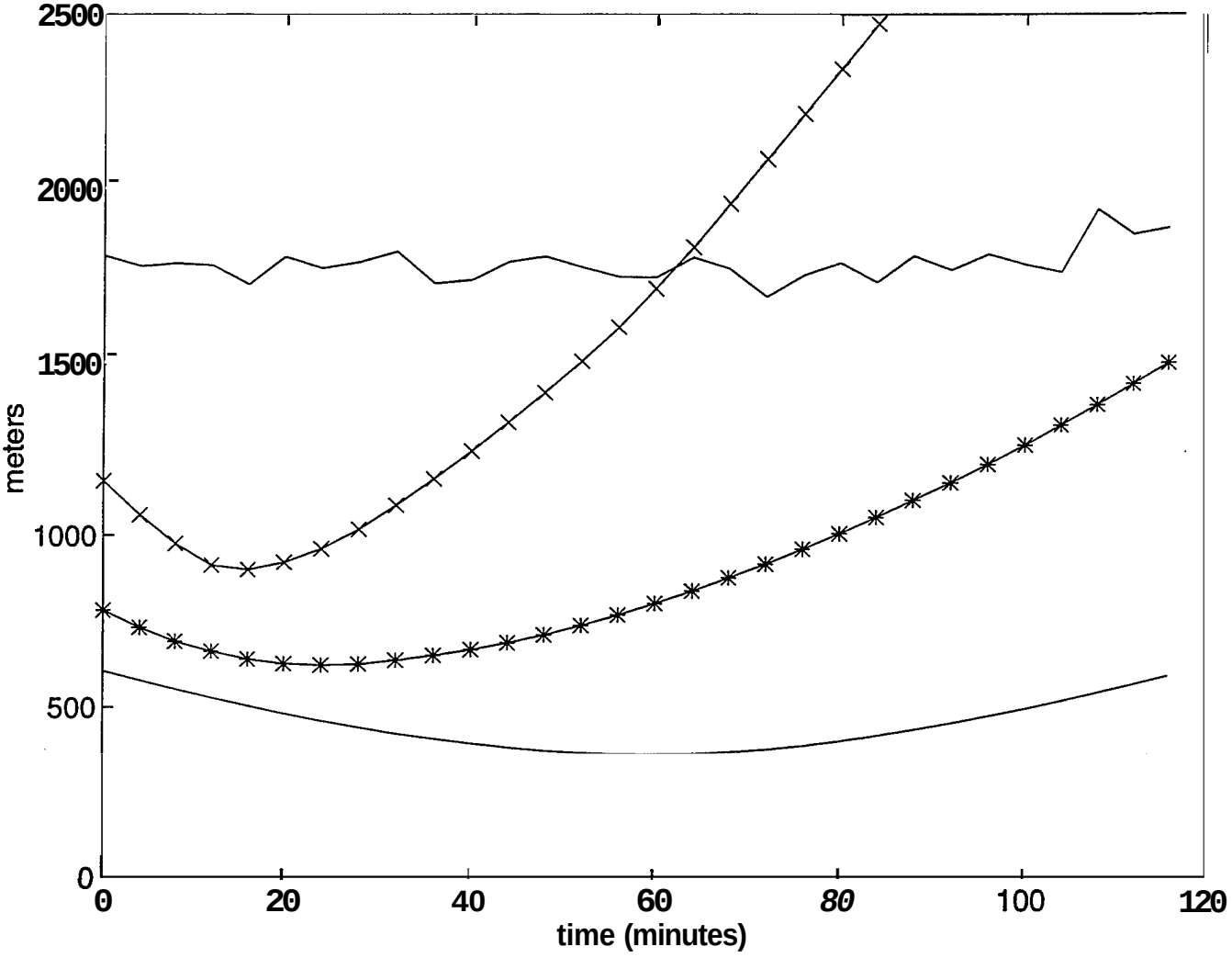


Figure 7: True Positions and Actual Target Measurements

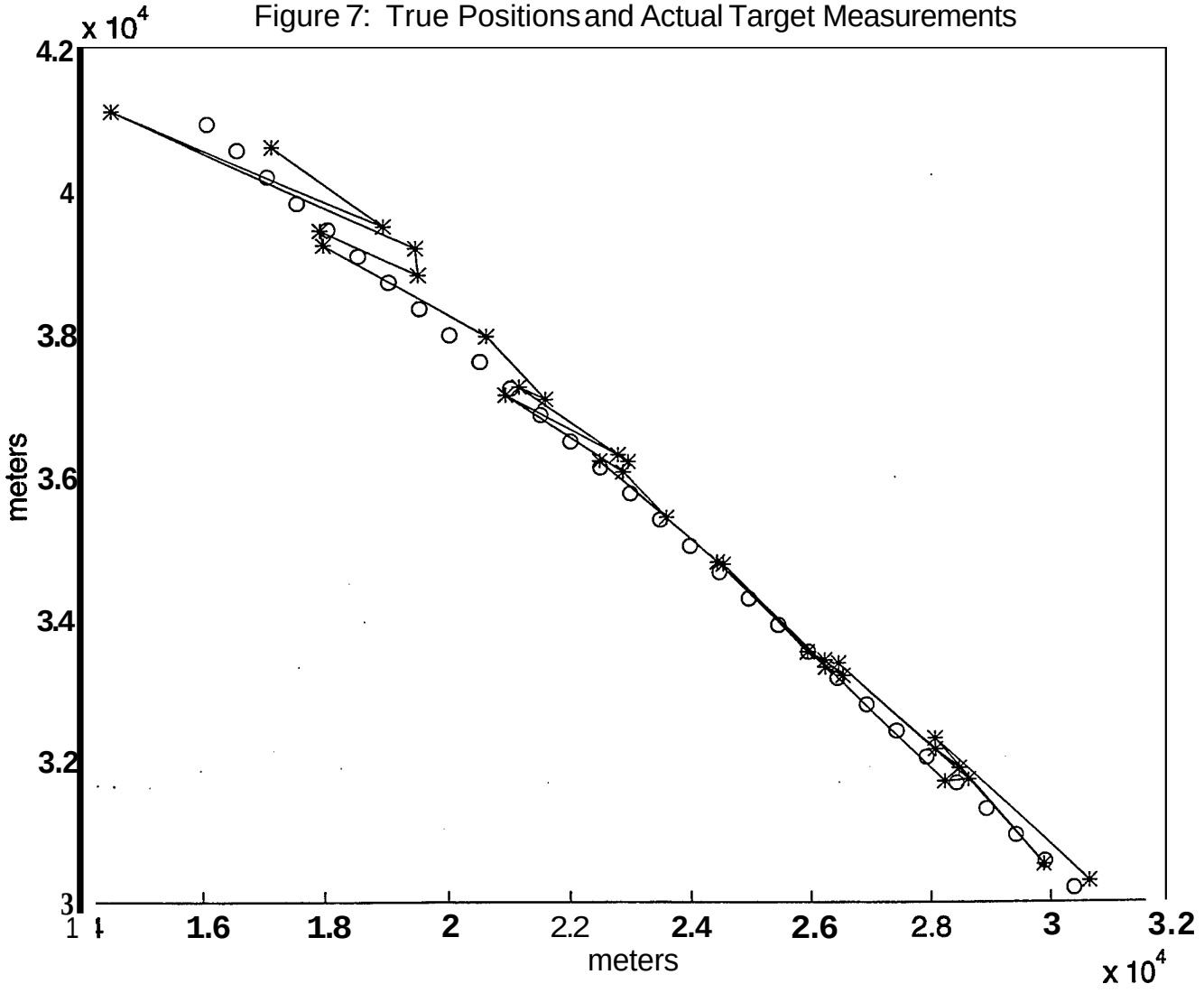


Figure 8: Typical Runaway Track is High Speed

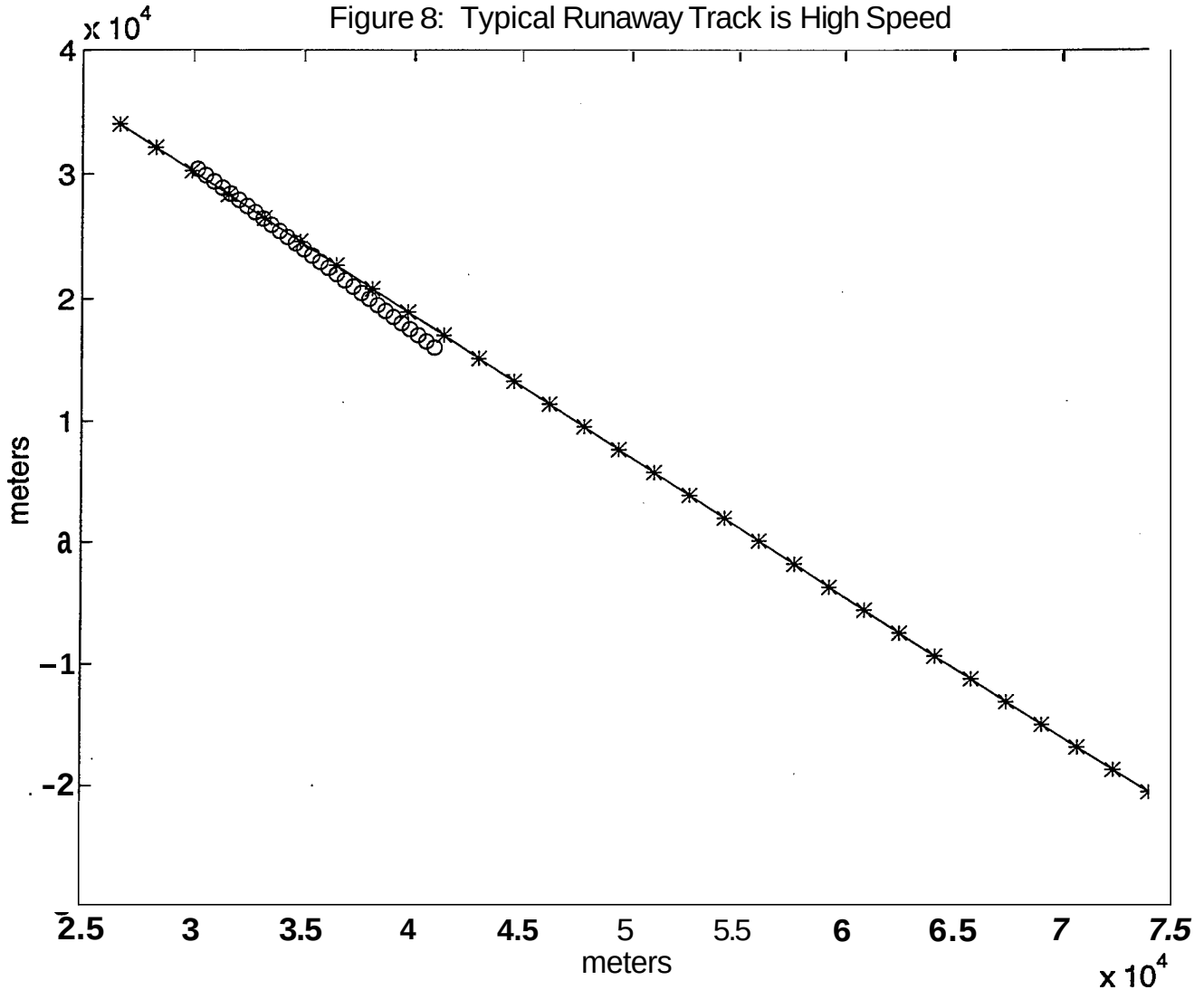


Figure 9: Min, Max, and Mean Number of Target Points in Best Track

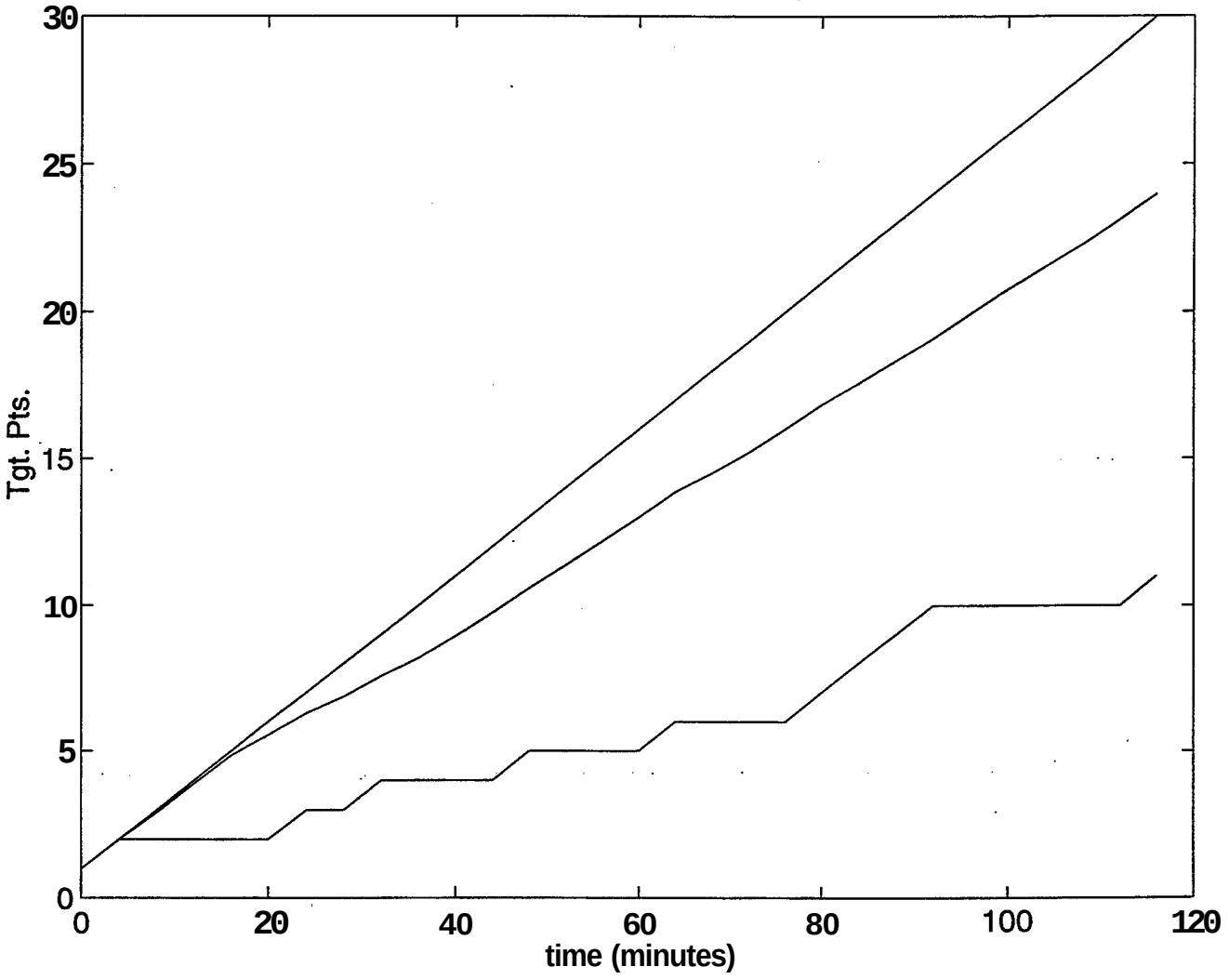


Figure 10: Mean MHT Distance Errors

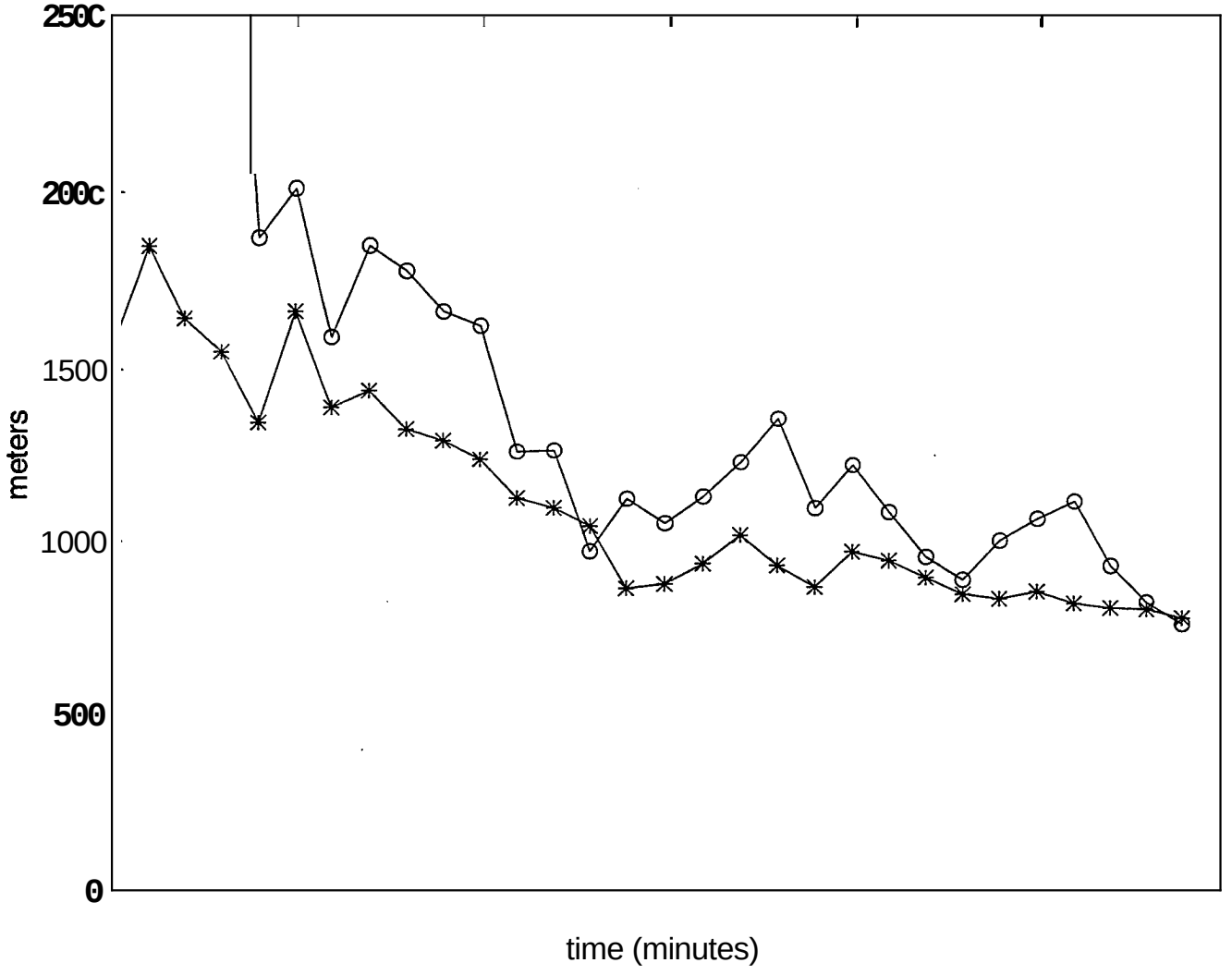


Figure 11: Min, Max, and Mean # Target Points in Best Track Prior to Turn

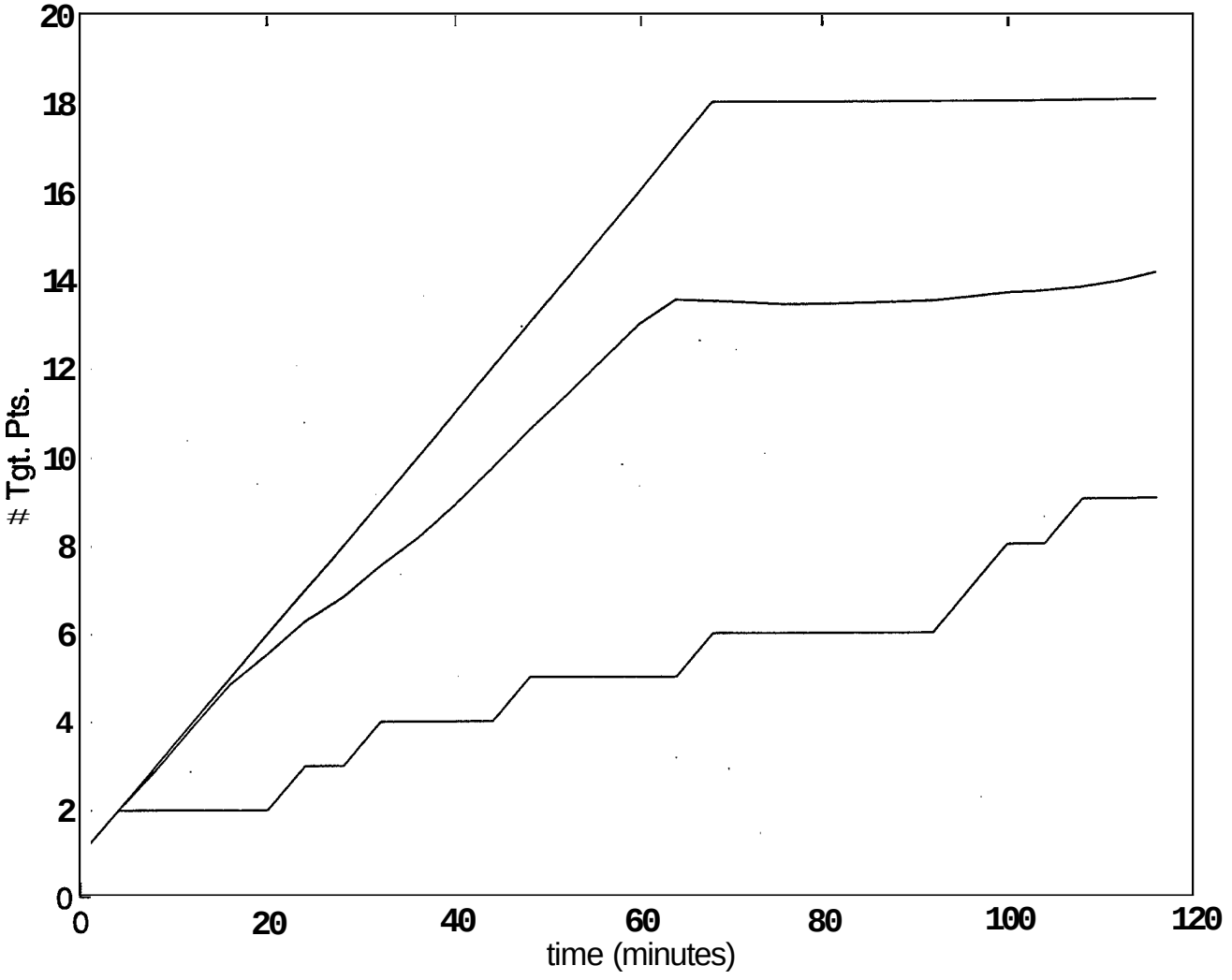


Figure 12: Mean # Target Points in Best Tracks Before and After Turn

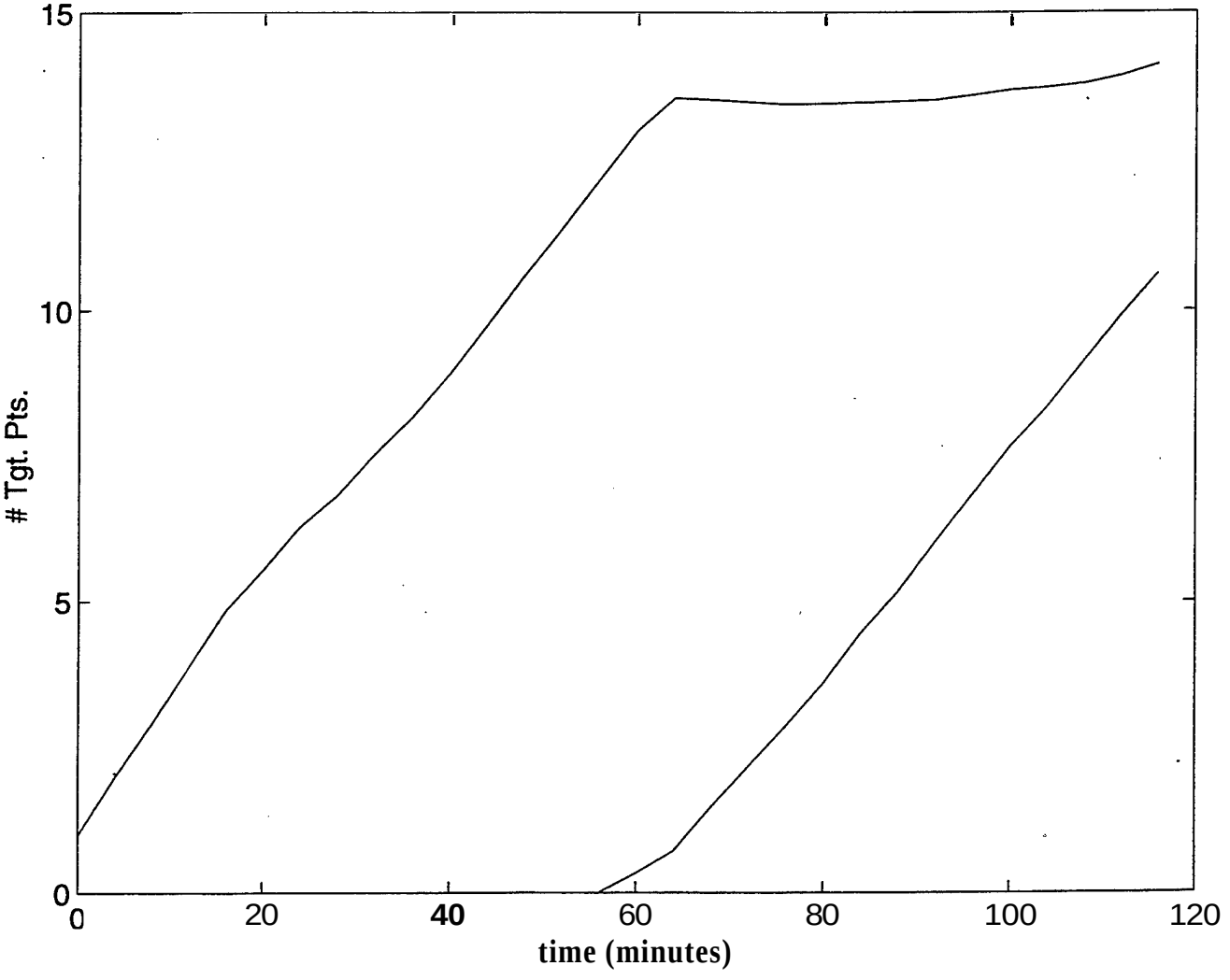


Figure 13: Mean Distance Errors for Turning Tracks – 5c

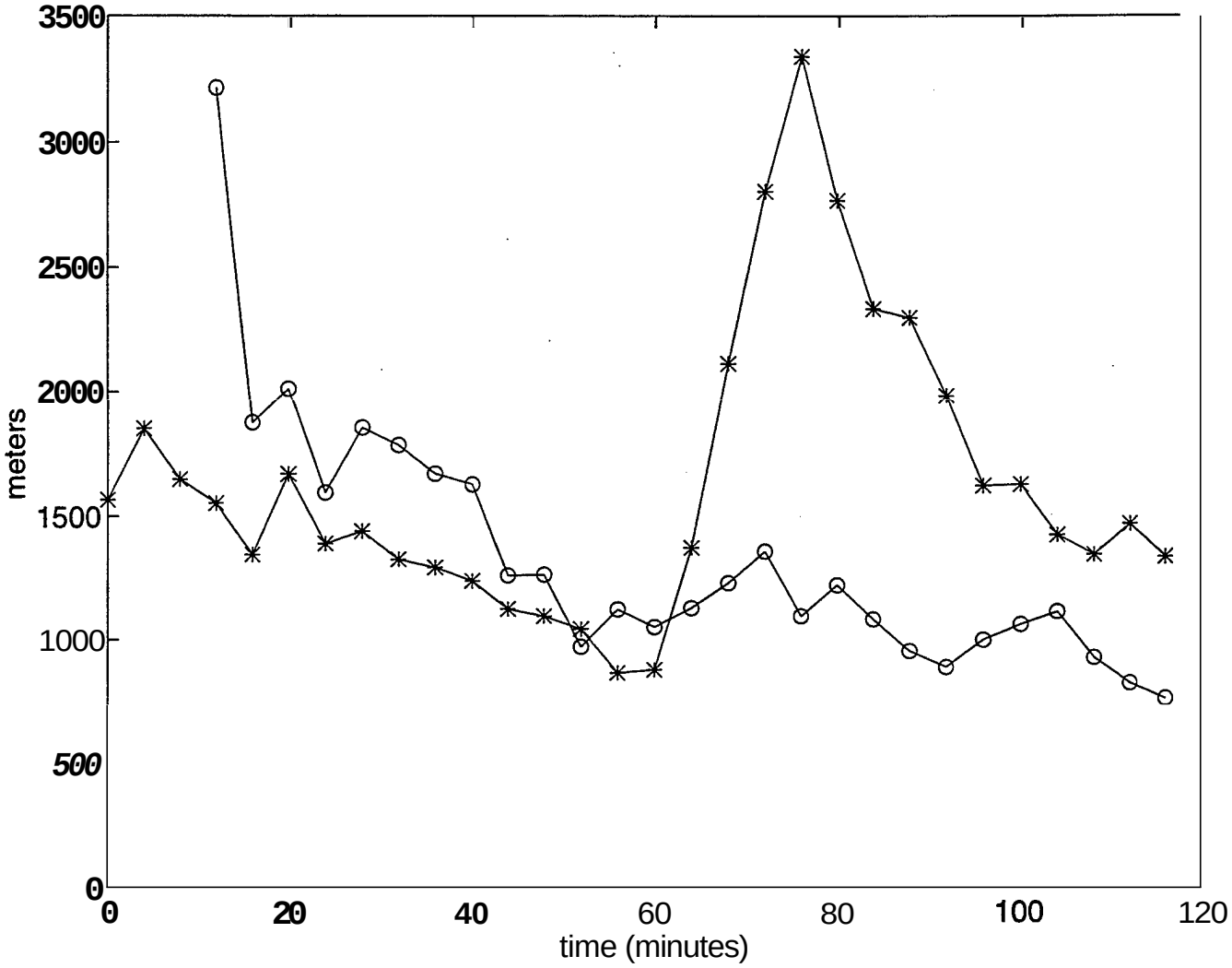


Figure 14: Algorithm Comparisons on Straight Track – 5c

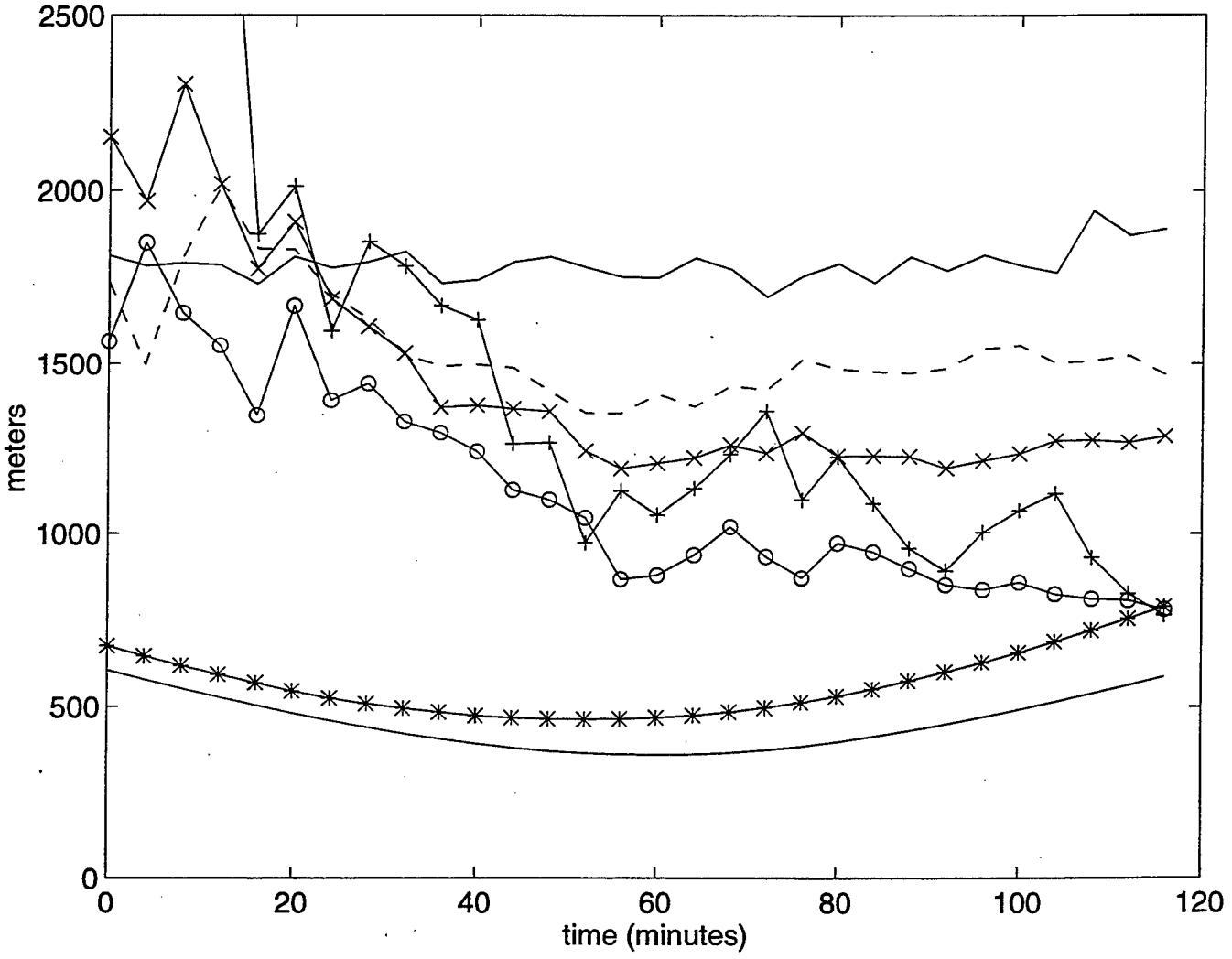


Figure 15: Algorithm Comparisons on Straight Track – 10c

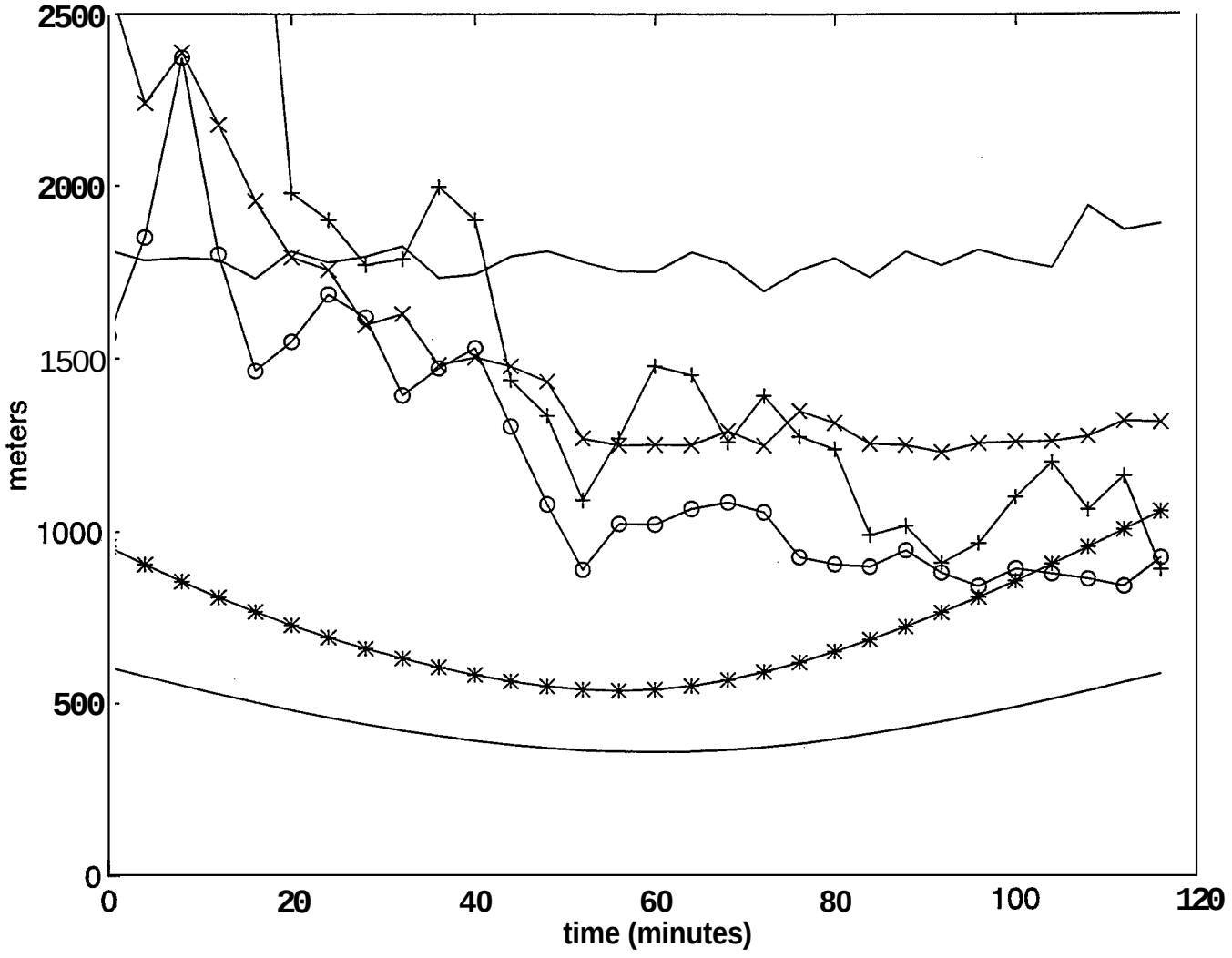
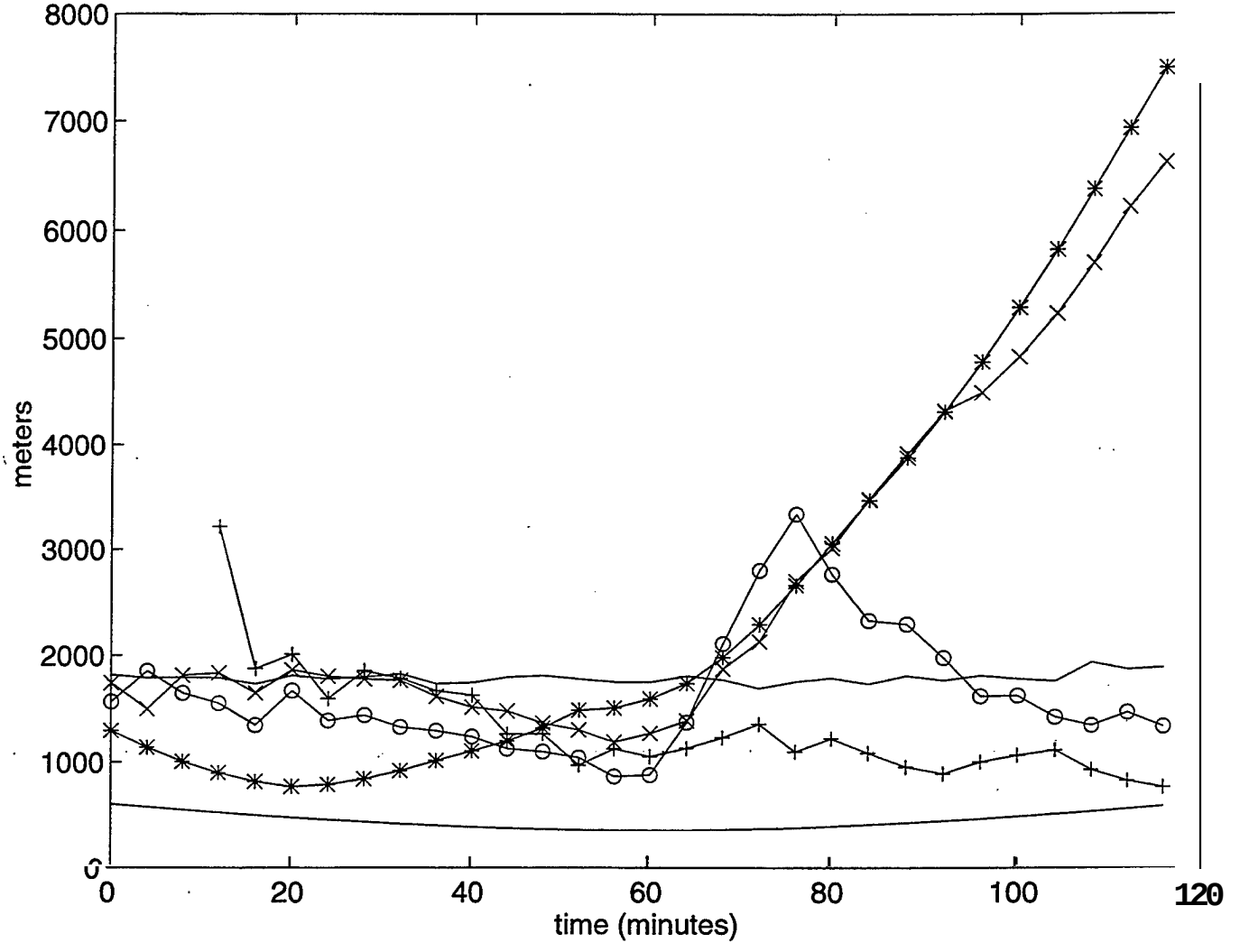


Figure 16: Mean Distance Errors for Turning Tracks – 5c



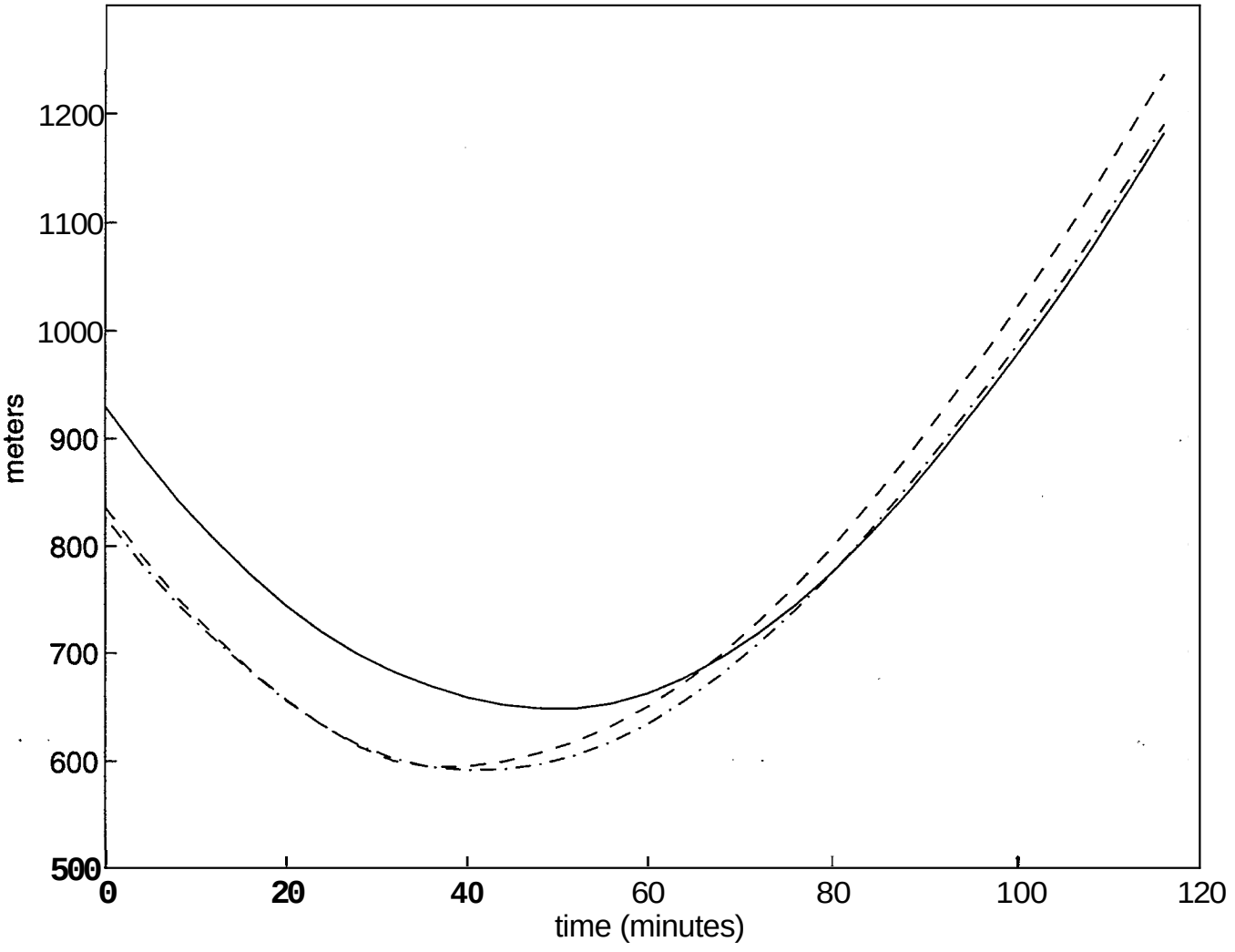

1200
1100
1000
900
800
700
600
500
meters
0
20
40
60
80
100
120
time (minutes)

Figure 18: Covariance Deflation Comparison – 10c

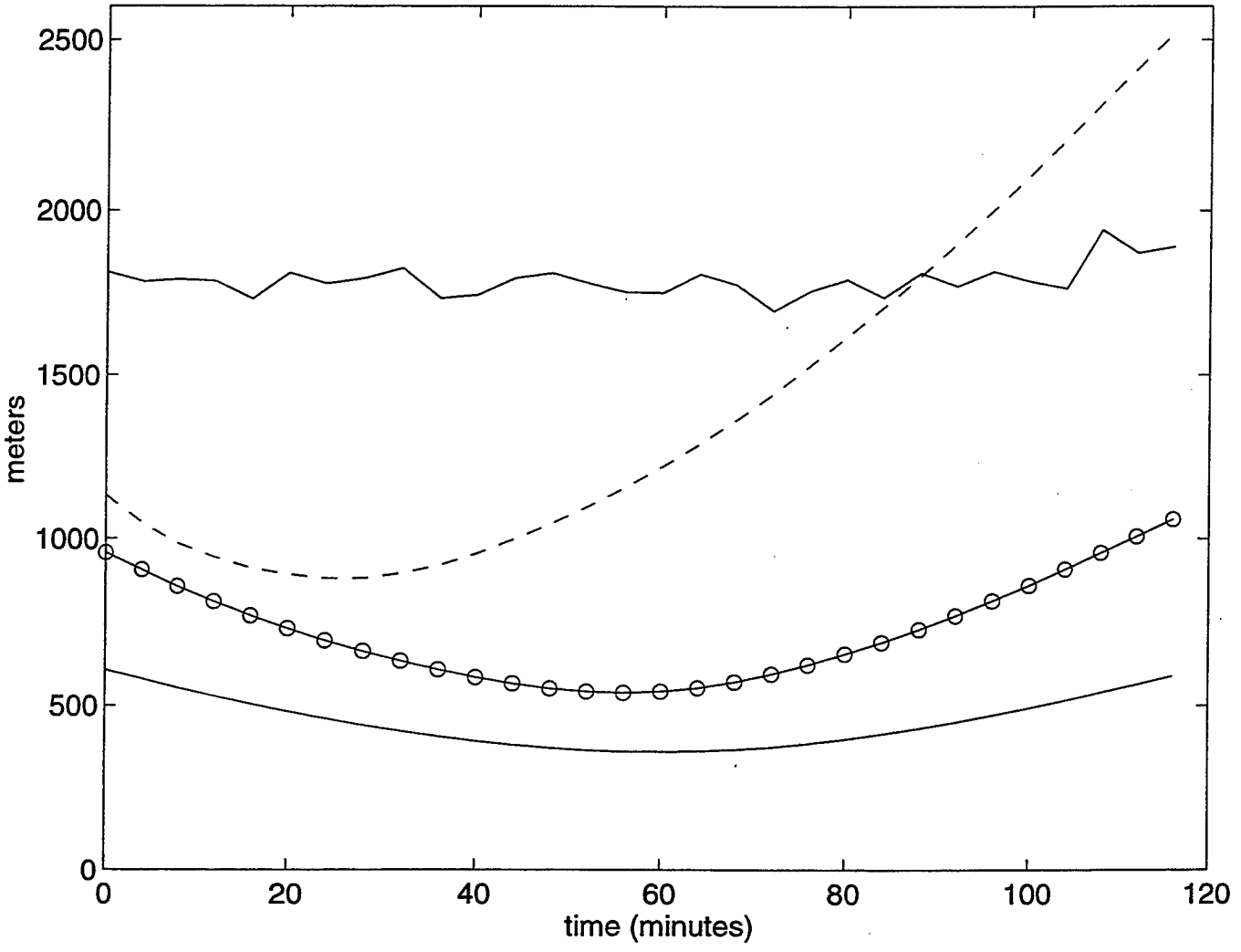


Figure 19: Algorithm Comparisons on Straight Track – 5c

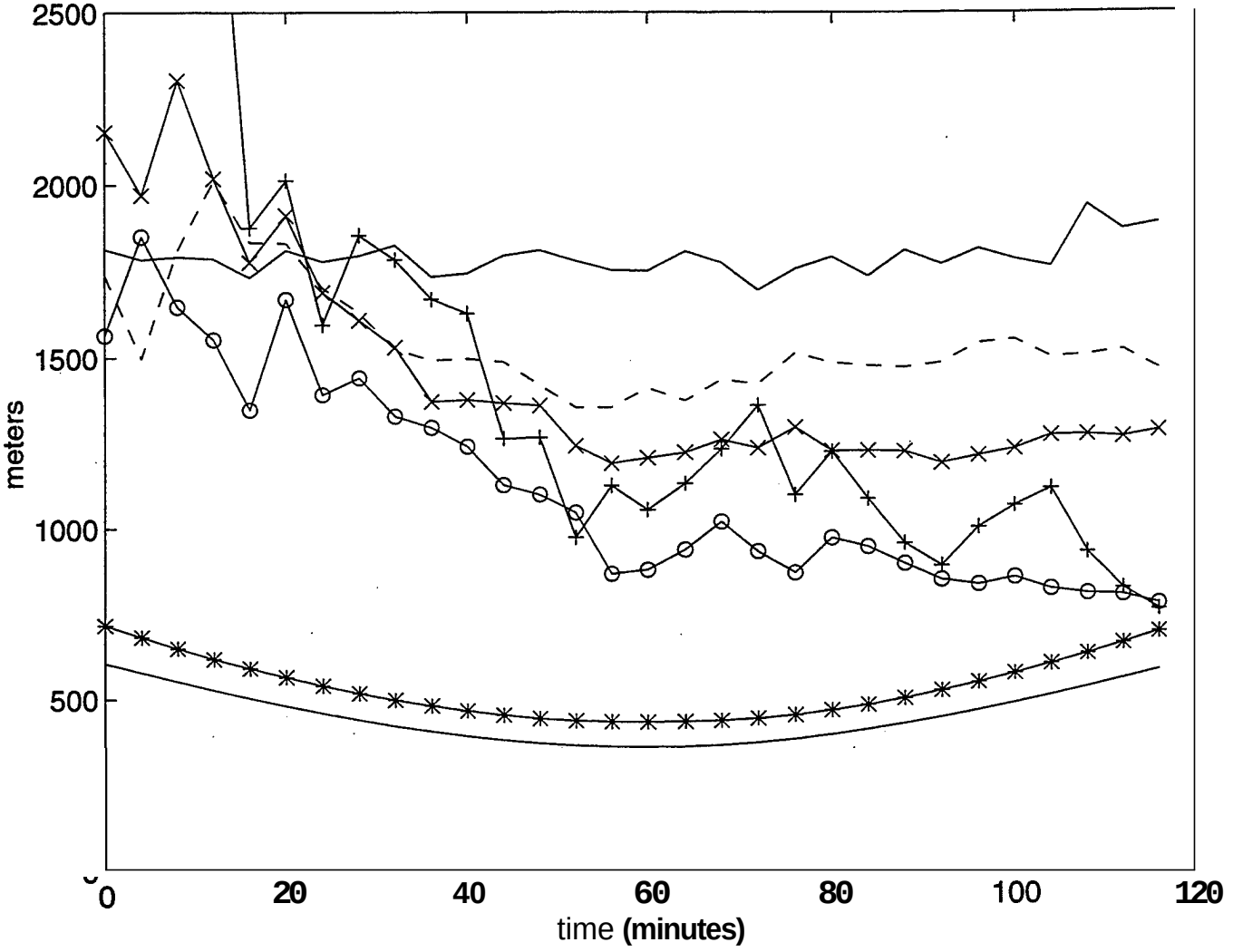


Figure 20: Algorithm Comparisons on Straight Track – 10c

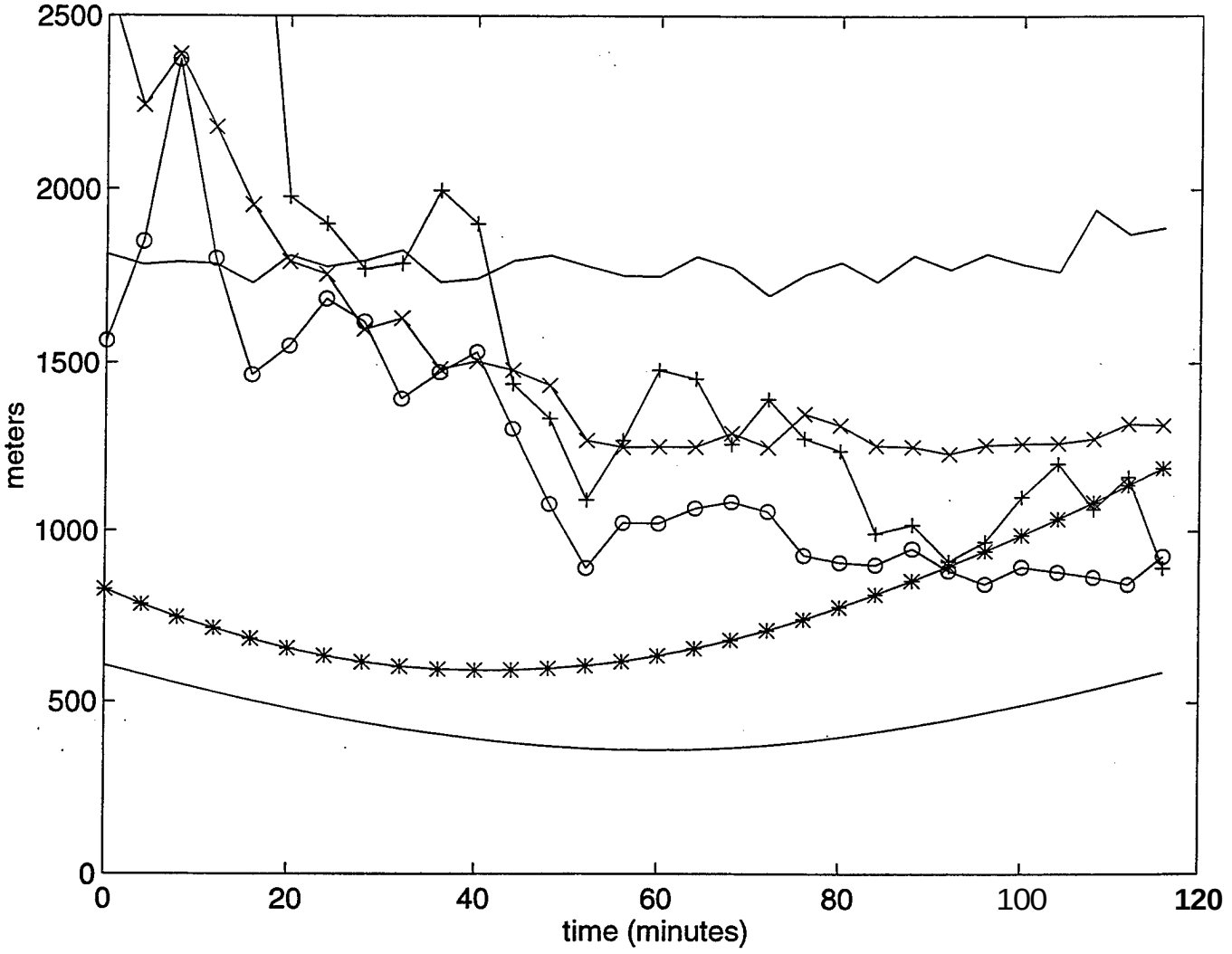


Figure 21: Mean Distance Errors for Crossing Tracks with Revised PMHT – 10c

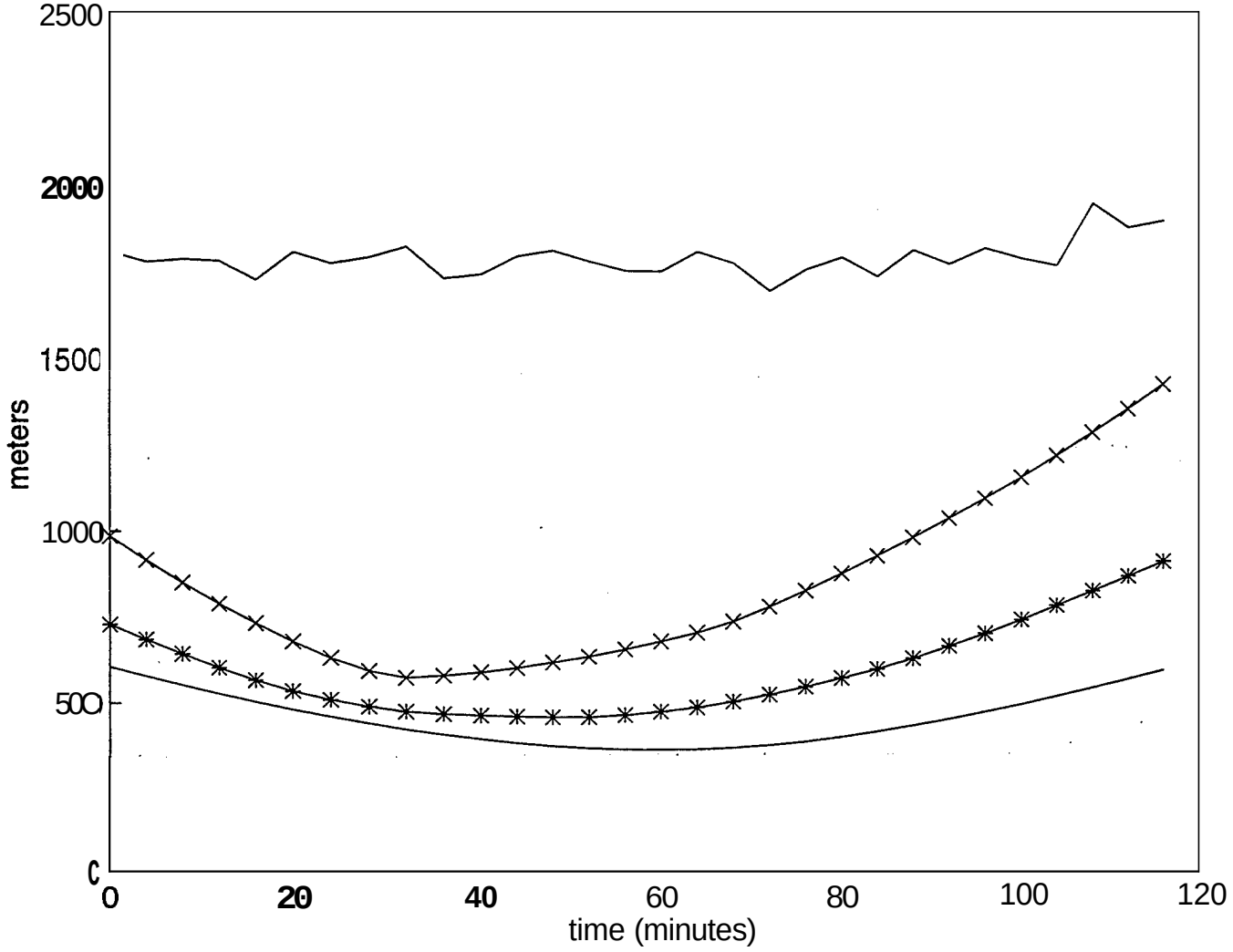


Figure 22: Mean Distance Errors for PMHT with Attribute Data – 10c

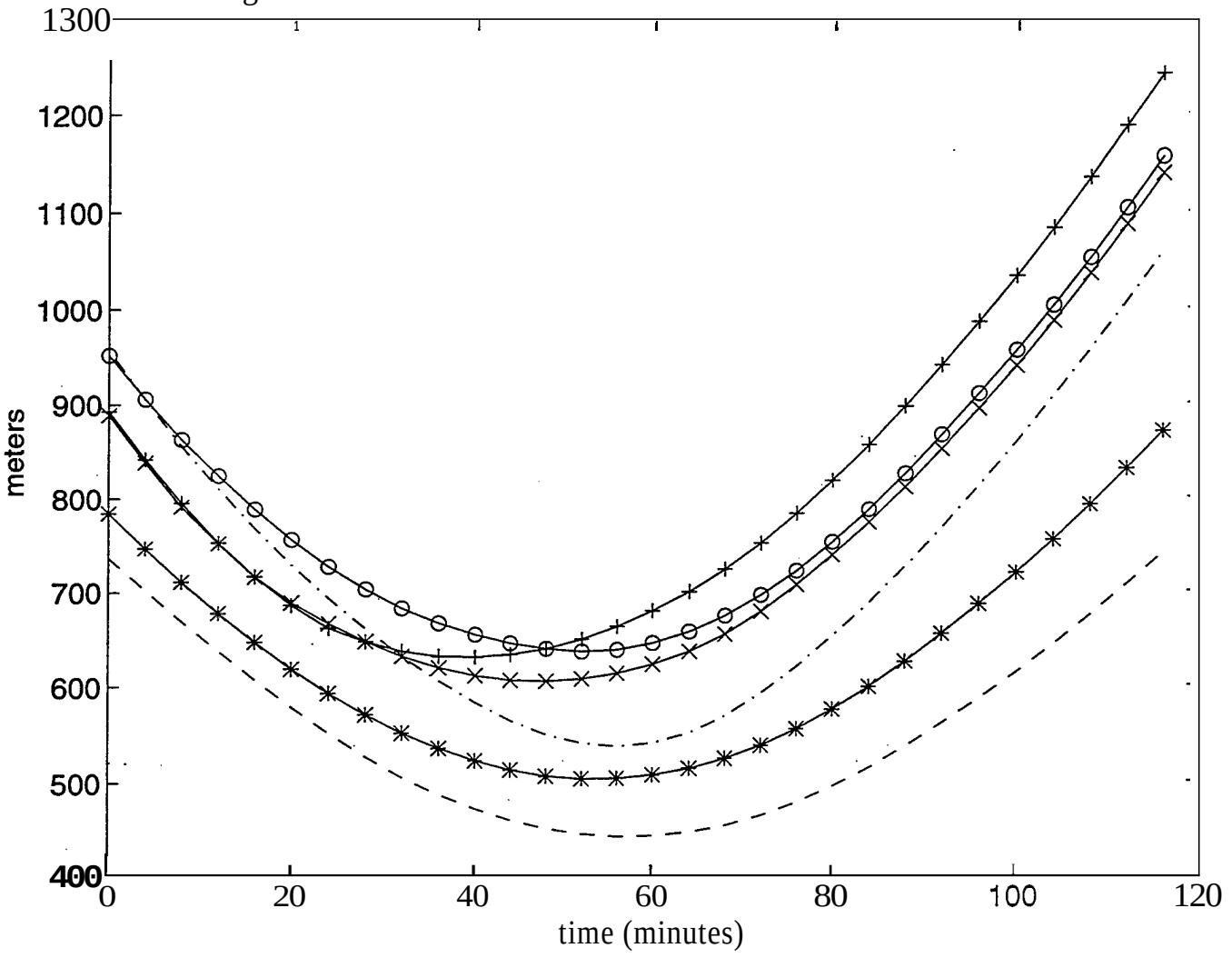


Figure 23: Mean Distance Errors with Attribute Data and Varying N–init – 5c

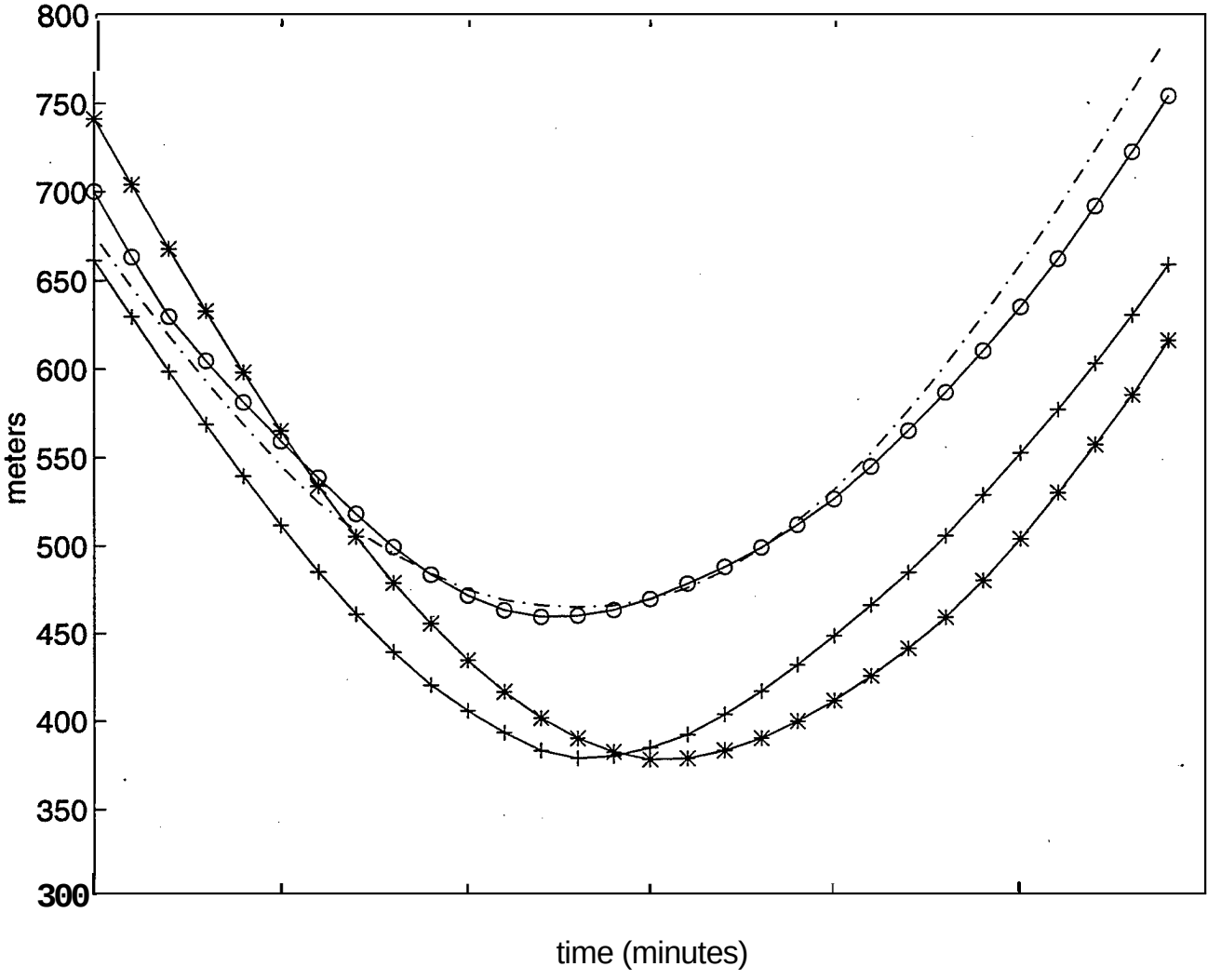
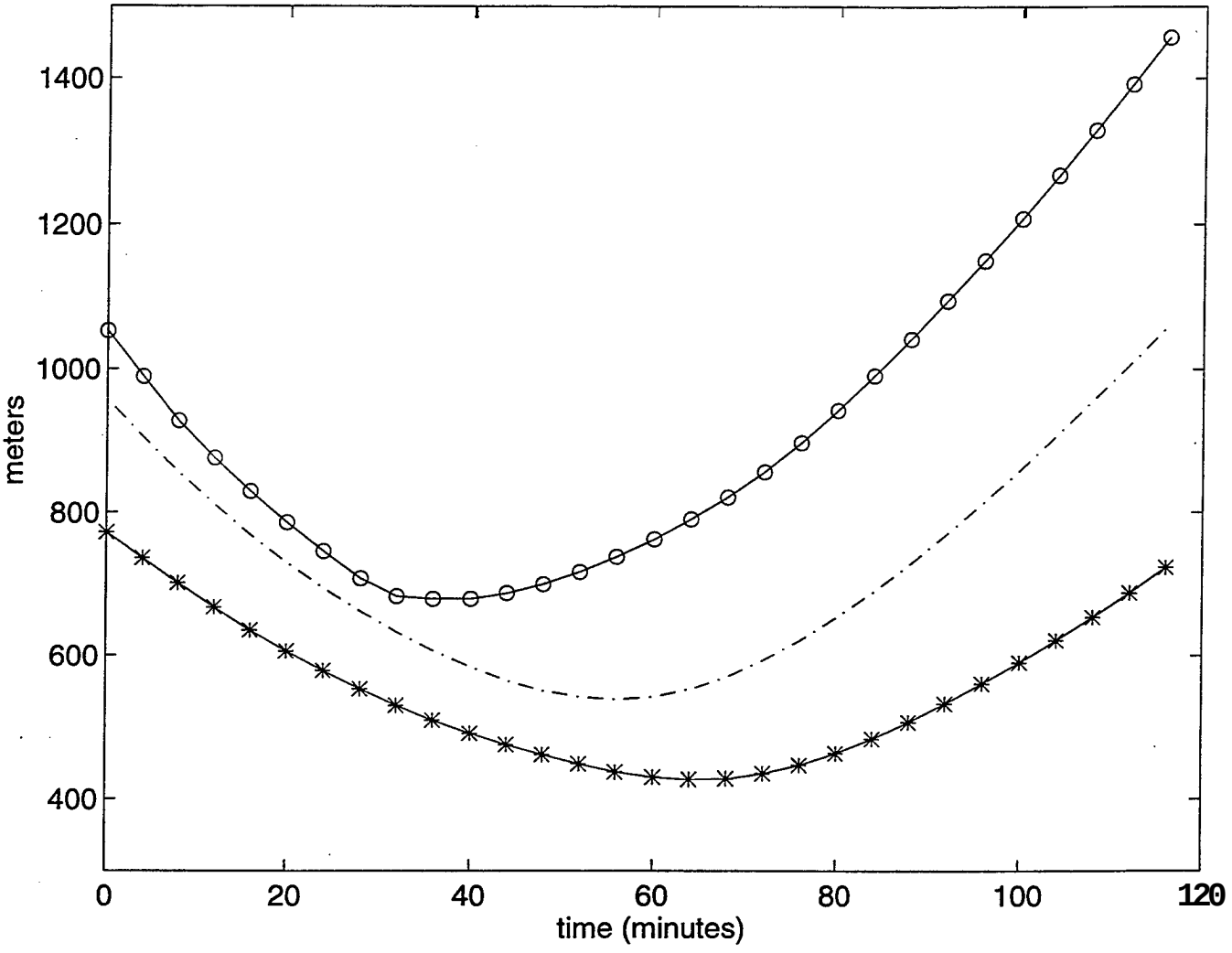


Figure 24: Mean Distance Errors with Attribute Data and Varying N–init – 10c



# INITIAL DISTRIBUTION LIST

| | | No. Copies |
|---|---|---|
| 1. | Defense Technical Information Center<br>8725 John J. Kingman Rd, STE 0944<br>Ft. Belvoir, VA 22060-6218 | 2 |
| 2. | Dudley **Knox** Library, Code 52<br>Naval Postgraduate School<br>411 Dyer Road<br>Monterey, CA 93943-5101 | 2 |
| 3. | Research Office, Code 09<br>Naval Postgraduate School<br>589 Dyer Road<br>Monterey, CA 93943-5138 | 1 |
| 4. | Chairman, Code EC<br>Department of Electrical and Computer Engineering<br>Naval Postgraduate School<br>833 Dyer Road<br>Monterey, CA 93943-5121 | 1 |
| 5. | Professor R.G. Hutchins, Code EC/Hu<br>Department of Electrical and Computer Engineering<br>Naval Postgraduate School<br>833 Dyer Road<br>Monterey, CA 93943-5121 | 2 |
| 6. | Mr. D.T. Dunham<br>Department of Electrical and Computer Engineering<br>Naval Postgraduate School<br>833 Dyer Road<br>Monterey, CA 93943-5121 | 2 |
| 7. | Naval Undersea Warfare Center<br>Attn: Dr. Roy Streit, Code 2002, Bldg. 1171<br>1176 Howell Street<br>Newport, RI 02841-5047 | 1 |