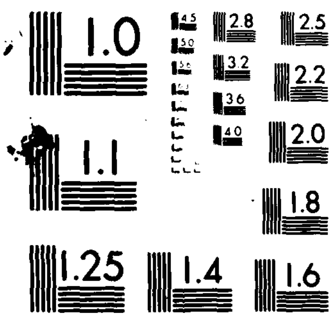
AD-A113 351    NAVAL RESEARCH LAB  WASHINGTON DC    F/G 15/4
AN IMPROVED PROCEDURE FOR ADAPTIVE ESTIMATION OF MANEUVERING IN--ETC(U)
APR 82  W W WILLMAN
UNCLASSIFIED    NRL-MR-4813    NL

END
DATE
FILMED
5 82
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0

AD-A113 351

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER<br>NRL Memorandum Report 4813 | 2. GOVT ACCESSION NO.<br>AD-A113 351 | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle)<br>AN IMPROVED PROCEDURE FOR ADAPTIVE ESTIMATION OF MANEUVERING IN A SHIP TRACKING ALGORITHM | | 5. TYPE OF REPORT & PERIOD COVERED<br>Interim report on a continuing NRL problem.<br>6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s)<br>W.W. Willman | | 8. CONTRACT OR GRANT NUMBER(s) |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>Naval Research Laboratory<br>Washington, DC 20375 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS<br>SSPO 11224N; J0092-SB; 79-1523-0-2 |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Naval Sea Systems Command<br>Washington, DC 20362 | | 12. REPORT DATE<br>April 21, 1982<br>13. NUMBER OF PAGES<br>36 |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) | | 15. SECURITY CLASS. (of this report)<br>UNCLASSIFIED<br>15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

| | |
|---|---|
| Ship tracking | Ocean surveillance |
| Command and control | Adaptive driving noise |
| Recursive filtering | Bearing-only tracking |
| Kalman filtering | |

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A modification is developed for both variants of the single-target ship tracking algorithm of NRL Memorandum Report 4579. The modification is an improvement in the adaptive procedure for estimating the amount of the ship's maneuvering from reports of its location, which can be bearing-only reports in one of the two variants. Tracking performance is improved most markedly when the report occurrances are highly irregular. Experimental FORTRAN implementations and numerical examples are given.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

CONTENTS



FIGURES

# AN IMPROVED PROCEDURE FOR ADAPTIVE ESTIMATION OF MANEUVERING IN A SHIP TRACKING ALGORITHM

## 1. INTRODUCTION

A single-target ship tracking algorithm has been developed in Refs. [1] - [3], and is also incorporated as a component of the multi-target tracker described in Ref. [4]. This algorithm operates on ship location reports which may occur sporadically in time, and adaptively estimates the intensity of the ship's maneuvering behavior from the report data rather than accepting it as a user-specified parameter. Maneuvering here is defined as deviations from a steady course, which is also estimated from the data. Ref. [3] also describes a variant of this algorithm which does not distinguish between down-track and cross-track maneuvering, but which can make more effective use of bearing - only reports, as opposed to reports of ship position which are meaningfully localized in two dimensions.

Another procedure has subsequently been devised for this maneuvering estimation which significantly improves the performance of both tracking algorithm variants. This new procedure is the subject of this report, and differs basically by the use of weighted averages of certain statistics, rather than simple averages, to form estimates of maneuvering intensity. The reliability of these statistics as measurements of maneuvering intensity varies with the spacing of adjacent reports. The new procedure gives relatively lower weight to the less reliable statistics in the case of unevenly spaced reports, resulting in more reliable maneuvering intensity estimates. Hence, the resulting improvement in tracking performance is most pronounced when the report occurrances are highly irregular. However, there seems to be some improvement even in the case of regularly spaced reports, as explained in Section 6 of this report.

## 2. PURPOSE AND ROLE OF IMPROVEMENT

For planar motion, the ship tracking algorithm of Refs. [1] - [3] is based on the idea of approximating the motion of the (single) ship being tracked as the vector sum of a constant (average) velocity and a two-dimensional (random) Brownian motion. The Brownian motion's intensity parameter, a 2x2 matrix Q in rectangular coordinates, is selected to correspond to the extent of maneuvering performed by the ship with respect to a constant-speed, constant-heading

**Manuscript submitted March 2, 1982.**

course (the average velocity). Both the average velocity and the maneuvering intensity matrix Q are treated as constant parameters to be estimated from the observed input data. This input consists of a time-ordered series of reports of ship location, not necessarily evenly spaced in time, such that each report specifies a time, the observed ship position at that time, and the (2x2) covariance matrix (or, equivalently, a containment ellipse) for the error in this observed position. An exception, however, is a modified tracking procedure described in Ref. [3], which allows some (or all) of these reports to specify only the observed ship bearing from some given point at that time, and the variance of the bearing observation error.

In either case, the tracking algorithm operates recursively in time, basically by propagating the ship track forward between observations by dead reckoning and updating it whenever a new report is received. The ship position and average velocity are treated as a four-component "state vector," for which a current conditional mean and covariance matrix are generated by a standard Kalman filter. Another recursive procedure is used to estimate the maneuvering intensity parameter Q from the "innovations" of the Kalman filter. These estimates are then used as "driving noise" parameters in the Kalman filter adaptively to modify its subsequent operation. The details of this process are described in Refs. [1] - [3], but can be summarized as the following recurring sequence of basic steps:

1. Upon the receipt of a new report of observed ship location, propagating the conditional probability distribution of the ship's "state vector" (position and velocity), given all <u>previous</u> data, to the current time. This distribution (conditioned on the same data) is generally available for the state at some earlier time, and is propagated from that time as if the value of Q estimated then were a precisely known parameter. The state distribution is treated as if it were (4-variate) Normal, so this step just amounts to propagating its mean and covariance matrix (14 independent components) to the current time with the standard Kalman filter for this value of Q and the motion model described above.

2. Updating the state probability distribution with the new observation, again using the standard Kalman filter for this case. This gives the conditional state distribution (specified by the mean and covariance matrix) given the current observation as well.

3. Updating the estimate of Q using the innovations (observed minus propagated position) from step 2.

4. Adjustment of some of the parameters of the updated position-velocity distribution from step 2 to compensate for the fact that the value of Q used in step 1 was (in general) different from the one just estimated in step 3.

Steps 1 and 4 are not performed for the first report. In this case, the track is initiated with an estimated Q of zero and a user-specified zero-mean circular Normal distribution for the average velocity. Also, step 1 can be performed to project a ship location distribution to a time at which there is no observation. This just amounts to an extended form of dead reckoning, in which an entire containment ellipse (representing the Normal distribution of the position components of the state vector) is propagated, not just a most likely position (the center of this ellipse). In step 3, it is assumed that the Brownian maneuvering motion consists of statistically independent components parallel and perpendicular to the average velocity, so only a down-track and a cross-track maneuvering intensity are estimated, and the resulting Q matrix is always diagonal when tranformed for coordinates aligned with the currently estimated average velocity vector. The intensiities of these two Brownian motion components are further restricted to be identical (isotropic maneuvering) in the variant of Ref. [3], which allows the inclusion of bearing-only data.

The improvement reported here is in the procedure for estimating the maneuvering intensity parameter Q in step 3. The advantage of this new procedure occurs chiefly when the observation times are quite unevenly spaced. The difficulty with the previous method is that, in both variants, equal weight is given to each member of the Kalman filter's innovation sequence (from step 2) in constructing the estimate of Q. However, the precision of these innovation statistics as measure of the Q components varies approximately as the inverse of the elapsed time since the last report, so their use in this way for unusually closely spaced reports often results in a less reliable estimate of Q than ignoring them entirely. The new estimation procedure is devised so that these innovation statistics are weighted according to their reliability as measures of the Q-components being estimated, thereby avoiding this problem. As with the former procedure, variants can be developed for both the case of independent down-track and cross-track maneuvering components and the case of isotropic maneuvering with bearing-only reports, as described in detail below.

Another possible benefit of the new Q-estimation procedure described here is that it seems to cause the overall tracking algorithm to create somewhat larger containment ellipses for projected ship locations than the former procedure, even for evenly spaced observation times (see Figs. 2 and 3). This would correct a reported tendency of the former method to result in erroneously small containment ellipses when used on realistic ship tracking data.

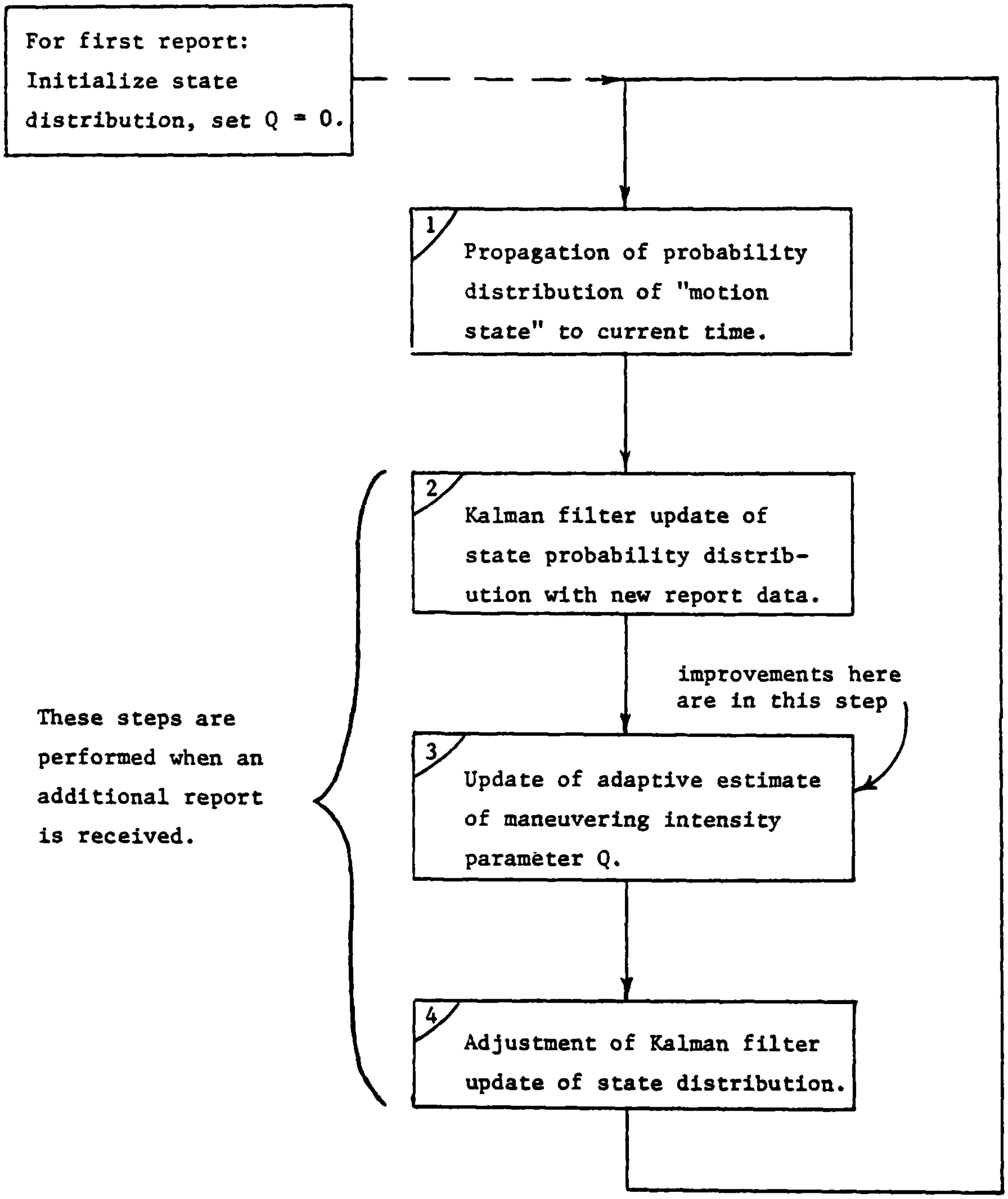



Fig. 1 — Outline of single-target ship tracking algorithm

## 3. RATIONALE FOR ONE-DIMENSIONAL SHIP MOTION

The new procedure for estimating the ship's maneuvering intensity parameter Q is based on a consideration of the corresponding one-dimensional case, in which the ship's motion is described by the (scalar) equation

$$\dot{x} = u + w$$

and the position observations at discrete time $t_i$ are

$$z_i = x(t_i) + n_i, \quad i = 0, 1, \ldots,$$

where the $n_i$ are independent zero-mean Normal random variables with known variances $r_i$, u is a constant but a-priori unknown average velocity, and w is a Normal white noise process with constant, but a-priori unknown variance parameter q. Tracking begins immediately after the initial observation $z_o$ at time $t_o$, at which point the two-vector

$$\begin{bmatrix} x(t_o) \\ \hline u(t_o) \end{bmatrix}$$

is regarded as having a bivariate Normal probability distribution with mean

$$\begin{bmatrix} z_o \\ \hline 0 \end{bmatrix}$$

and covariance matrix $\left[\begin{array}{c|c} r_o & 0 \\ \hline 0 & \lambda^2 \end{array}\right]$ ,

where $\lambda^2$ is some a-priori specified ship speed variance.

Now consider the approximation that, at observation time $t_i$, the conditional probability densities of q and of the vector $\begin{bmatrix} x(t_i) \\ \hline u \end{bmatrix}$, given the observations $z_o, \ldots, z_i$, are statistically independent such that,

$$q \sim \text{Normal}\ (\bar{q},\ s)$$

and

$$\begin{bmatrix} x(t_i) \\ \hline u \end{bmatrix} \sim \text{Normal} \left( \begin{bmatrix} \hat{x}_i \\ \hline \hat{u}_i \end{bmatrix} , \begin{bmatrix} p_{xx} & p_{xu} \\ \hline p_{xu} & p_{uu} \end{bmatrix} \right).$$

For the next observation time $t_{i+1}$, define

$$\tau = t_{i+1} - t_i , \tag{1}$$

$$a = p_{xx} + 2\tau p_{xu} + \tau^2 p_{uu}, \tag{2}$$

$$m = a + \bar{q}\tau, \tag{3}$$

$$\bar{x} = \hat{x}_i + \hat{u}_i \tau, \tag{4}$$

and

$$\varepsilon = z_{i+1} - \bar{x}, \tag{5}$$

and denote $x(t_{i+1})$ by x and $r_{i+1}$ by r for brevity. Then it follows from standard results for moments of Normal random variables that, given $z_o,\ldots,z_i$,

$$E(\varepsilon^2) = a + r + \bar{q}\tau , \tag{6}$$

$$\text{var}(\varepsilon^2) = 2\,(m+r)^2 + 3s\tau^2, \tag{7}$$

and

$$\text{cov}\,(q,\varepsilon^2) = s\tau. \tag{8}$$

By assumption, $E(q/z_o,\ldots,z_i) = \bar{q}$ (9)

and

$$\text{var}(q/z_o,\ldots,z_i) = s. \tag{10}$$

If at this point the (different) approximation is made of treating q and $\varepsilon^2$ as bivariate Normal random variables under this conditioning, with the preceding mean and covariance matrix elements, then it is a standard result that

$$E(q/\varepsilon^2) \triangleq \hat{q} = E(q) + \frac{\text{cov}(q,\varepsilon^2)}{\text{var}(\varepsilon^2)}\,[\varepsilon^2 - E(\varepsilon^2)] \tag{11}$$

and

$$\text{var}(q/\varepsilon^2) \triangleq \sigma = \text{var}(q) - \frac{\text{cov}^2(q,\varepsilon^2)}{\text{var}(\varepsilon^2)} , \tag{12}$$

where the conditioning on $z_o,\ldots,z_i$ is suppressed in the notation. Substituting Eqs. (6) - (10) into (11) and (12) we obtain

$$\hat{q} = \bar{q} + \frac{s}{2(\frac{m+r}{\tau})^2 + 3s}\left(\frac{\varepsilon^2 - (a+r)}{\tau} - \bar{q}\right) \tag{13}$$

and

$$\sigma = s - \frac{s^2}{2(\frac{m+r}{\tau})^2 + 3s} \quad . \tag{14}$$

Reducing the denominators of Eqs. (13) and (14) to "$2(m/\tau)^2 + s$" would convert these equations to the form of the updating step in the standard Kalman filter for constant q with

$$q - \text{measurement} = \frac{\varepsilon^2 - (a+r)}{\tau}$$

and

$$\text{measurement noise variance} = 2(m/\tau)^2$$

at measurement epoch i+1. This change is justified to some extent by the fact that m really depends via (2) and (3) on the previous estimates of q, which are used to compute $p_{xx}$. Also, the resulting Kalman filter estimate of q reduces to simply the average of the quantities

$$\frac{\varepsilon^2 - (a+r)}{\tau}$$

observed at each updating time when m,r and $\tau$ are all equal at each step, which is essentially the former estimation procedure for q described in Ref. [1], and is consistent with the justification given there as a least-squares estimation procedure. For unevenly spaced observations, however, this new updating procedure has the desirable property of estimating q as a weighted average of these quantities, the weight being less for epochs at which $\tau$ is small and the variance of the "q - measurement" is therefore large. Thus, it gives a sort of weighted least - squares estimate of q when used sequentially in the overall ship tracking algorithm. Rearranging terms gives this Kalman filter update step as

$$\hat{q} = \bar{q} + \frac{s\tau}{2\,m^2 + s\tau^2}\,[\epsilon^2 - (m+r)] \qquad (15)$$

and

$$\sigma = \frac{2\,m^2}{2\,m^2 + s\tau^2}\;s. \qquad (16)$$

Since it doesn't make sense to estimate maneuvering intensity with fewer than three observations, such updating is started on the third observation (for which $\tau > o$) with $s = \infty$, which corresponds to using $\bar{q} = o$ and replacing the "filter gain"

$$\frac{s\tau}{2\,m^2 + s\tau^2}$$

by $\frac{1}{\tau}$ in Eq. (15), and using $\sigma = 2\,(m/\tau)^2$ for Eq. (16).

A standard Kalman filter for estimating a constant q would use the value of $\sigma$ from one update time as the value of s for the next update. Since the approximate value for q can really change over extended periods of time, however, it is more reasonable to postulate a "forgetting time constant" T and use exponential deweighting between update times by computing "s" from the preceding "$\sigma$" as

$$s = e^{-\frac{\tau}{T}}\left[\sigma_{preceding} - 2\left(\frac{m}{t_{i+1} - t_o}\right)^2\right] + 2\left(\frac{m}{t_{i+1} - t_o}\right)^2 \qquad (17)$$

to avoid "locking in" to a value of $\hat{q}$. This particular deweighting scheme corresponds to q changing between the current and preceding observation times according to

$$\dot{q} = -\frac{q}{2T} + \text{white noise}, \qquad (18)$$

where the white noise intensity is such that the steady-state variance of q is $2\,\frac{m}{t_{i+1} - t_o}$, the value it would have if initialized at the current update time.

The estimate $\hat{q}$ here can assume negative values, whereas q must by definition be nonnegative. Hence the latest value of

$$\max(\hat{q}, o)$$

is always used as the maneuvering intensity parameter in the other parts of the overall ship tracking algorithm.

As a last refinement, if the quantity

$$\alpha = \max\{o, \max(\hat{q}, o) - \max(\bar{q}, o)\} \tag{19}$$

is positive, it is assumed, as explained in Ref. [3], that $p_{uu}$ (which is computed at preceding update time) should have been larger by approximately $\frac{\alpha}{t-t_o}$, where t is the current time. This would increase $\sigma$ by

$$\frac{2}{\tau^2}(\bar{m}^2 - m^2) = \frac{2}{\tau^2}(m+\bar{m})(\bar{m}-m)$$

if it were being initialized at this point. In the above $\bar{m}$ is the value of m which would result from (2) and (3) if $p_{uu}$ were increased as described, i.e.,

$$\bar{m} = m + \frac{\alpha\tau^2}{t-t_o} .$$

To compensate for having reduced the effects of the correlation between q and m from those in (13) and (14), the quantity $\alpha\tau$ is also added to $\bar{m}$, giving

$$\bar{m} = m + \alpha\tau(1 + \frac{\tau}{t-t_o}) ,$$

which would increase an initialized value of $\sigma$ by

$$\delta\sigma = 2[2\ m + \alpha\tau(1 + \frac{\tau}{t-t_o})]\ \frac{\alpha}{\tau}(1+ \frac{\tau}{t-t_o}) .$$

To be conservative and to avoid dividing by $\tau$, we therefore add the quantity

$$\delta\sigma = 2[2\ m + \alpha\tau]\ \frac{\alpha}{t-t_o} \tag{20}$$

to the result of Eq. (16) at <u>all</u> update times as an approximate correction for having ignored in the development of (16) some of the interactions between estimating q and estimating x and u in the overall ship tracking algorithm, namely the dependence of $p_{uu}$ on q. Of course, the other corrections of this sort in the overall algorithm, which are for parameters of the conditional distribution of x and u as described in Ref. [3], are retained as before.

## 4. PROCEDURE FOR TWO-DIMENSIONAL MOTION

Only planar motion in rectangular coordinates is discussed here. The algorithms for tracking such motion can easily be extended to tracking on a sphere, as described in Refs. [1] and [2]. In the context of planar motion,

$x$ = ship position

$v$ = ship velocity (average), and

$z_i$ = observed position at time $t_i$, $i = 0,1,\ldots.$

are all 2-vectors with components in these two coordinates, and the composite "motion state" vector $\begin{bmatrix} x \\ v \end{bmatrix}$ has four components. The conditional distribution of this state vector is approximated by the tracking algorithm as 4-variate Normal, whose mean and covariance matrix are denoted here in bivariate partitions as

$$\begin{bmatrix} \hat{x}(t) \\ \hline \hat{v}(t) \end{bmatrix} \text{ and } \left[\begin{array}{c|c} P_1(t) & P_2(t) \\ \hline P_2^T(t) & P_3(t) \end{array}\right] \text{ for generic time t.}$$

For convenience, we also denote the current estimate of the maneuvering intensity parameter as Q and the covariance matrix of the error in the i-th observation as $R_i$, both of which are 2x2 symmetric positive semi-definite matrices. Also, $\hat{q}_d(i)$ and $\hat{q}_c(i)$ are used to denote interim estimates of the (scalar) down-track and across-track maneuvering intensities created immediately after the i-th observation, and $\sigma_d$ and $\sigma_c$ to denote corresponding variance parameters, in accordance with the notation of the preceding section.

The operation of the overall ship tracking algorithm, with the improved adaptive estimation procedure for maneuvering, can be summarized as follows for the case in which the reports all specify a position which is localized in both dimensions:

Initialization

Upon the receipt of the initial report $(z_o, R_o)$ at time $t_o$, tracking is started with

$$\hat{x}(t_o^+) = z_o \qquad \text{(2 - vector)}$$

$$P_1(t_o^+) = R_o \qquad \text{(2x2 matrix)}$$

$$P_2(t_o^+) = 0 \qquad \text{(2x2 matrix)}$$

$$P_3(t_o^+) = \frac{1}{2}\left[\begin{array}{c|c} \lambda^2 & 0 \\ \hline 0 & \lambda^2 \end{array}\right] \quad ; \quad \lambda^2 \text{ a user-specified prior speed variance}$$

$$\hat{q}_d(o) = \hat{q}_c(o) = 0 \quad \text{(or some other user-specified positive value)}$$

From time $t_i^+$ to time $t_{i+1}^+$ ; $i = 0,1,...$

Tracking proceeds by performing the summary steps of Fig. 1 as follows, where "=" denotes a replacement operation as in FORTRAN.

Step 1 - Propagate state distribution to $t_{i+1}^+$ by successively computing:

$$c_d = \max\{\hat{q}_d(i), 0\}$$

$$c_c = \max\{\hat{q}_c(i), 0\}$$

$$\theta = \tan^{-1}\left(\frac{\hat{v}_1(t_i^+)}{\hat{v}_2(t_i^+)}\right) \quad ; \quad \hat{v}_1 \text{ and } \hat{v}_2 \text{ are the two components of } \hat{v} \text{ (usually local east and north components of velocity estimate)}$$

$$q_{11} = c_d \cos^2\theta + c_c \sin^2\theta$$

$$q_{12} = (c_d - c_c) \sin\theta \cos\theta$$

$$q_{22} = c_d \sin^2\theta + c_c \cos^2\theta$$

$$Q = \left[\begin{array}{c|c} q_{11} & q_{12} \\ \hline q_{12} & q_{22} \end{array}\right] \qquad \text{(2x2 matrix)}$$

$$\tau = t_{i+1} - t_i$$

$$\bar{x} = \hat{x}(t_i^+) + \hat{v}(t_i^+)\,\tau \qquad \text{(2 - vector)}$$

$$\left.\begin{aligned} A &= P_1(t_i^+) + [P_2(t_i^+) + P_2^T(t_i^+)]\tau + P_3(t_i^+)\tau^2 \\ M_1 &= A + Q\tau \\ M_2 &= P_2(t_i^+) + P_3(t_i^+)\tau \\ M_3 &= P_3(t_i^+) \end{aligned}\right\} \text{2x2 matrices}$$

At this point, the conditional distribution of the state at time $t_{i+1}$, given $z_o,\ldots z_i$, can be approximated as (4-variate) Normal with

$$\text{mean} = \begin{bmatrix} \bar{x} \\ \hline \hat{v}(t_i^+) \end{bmatrix}$$

and

$$\text{covariance} = \left[\begin{array}{c|c} M_1 & M_2 \\ \hline M_2 & M_3 \end{array}\right] ,$$

and $t_{i+1}$ need not be an actual observation time for this purpose, but may be any time after $t_i$ for which the projected probability distribution is desired.

Step 2 - Update state distribution with new report:

Let z and R denote $z_{i+1}$ and $R_{i+1}$ henceforth. Compute

$$\left.\begin{aligned} \hat{x}(t_{i+1}^+) &= \bar{x} + M_1\,(M_1 + R)^{-1}(z - \bar{x}) \\ \hat{v}(t_{i+1}^+) &= \hat{v}(t_i^+) + M_2^T\,(M_1 + R)^{-1}(z - \bar{x}) \end{aligned}\right\} \text{2-vectors}$$

$$\left.\begin{aligned} P_1(t_{i+1}^+) &= M_1 - M_1(M_1 + R)^{-1}M_1 \\ P_2(t_{i+1}^+) &= M_2 - M_1(M_1 + R)^{-1}M_2 \end{aligned}\right\} \qquad \text{2x2 matrices}$$

$$P_3(t_{i+1}^+) = M_3 - M_2(M_1 + R)^{-1}M_2$$

Step 3 - Update of maneuvering estimate.

For i=0,1 only: set $\hat{q}_d\ (i+1) = \hat{q}_d\ (i)$

and $\hat{q}_c\ (i+1) = \hat{q}_c\ (i)$.

Otherwise, compute in turn:

$$\gamma = t_{i+1} - t_o$$

$$\begin{bmatrix} b_d \\ \hline b_c \end{bmatrix} = \Omega M_1 \Omega^T \text{ and } \begin{bmatrix} r_d \\ \hline r_c \end{bmatrix} = \Omega R \Omega^T;\ \Omega = \left[\begin{array}{c|c} \cos\theta & \sin\theta \\ \hline -\sin\theta & \cos\theta \end{array}\right]$$

$$\begin{bmatrix} \varepsilon_d \\ \hline \varepsilon_c \end{bmatrix} = \left[\begin{array}{c|c} \cos\theta & \sin\theta \\ \hline -\sin\theta & \cos\theta \end{array}\right] (z - \bar{x})$$

$$\left.\begin{aligned} \sigma_d\ (3) &= 2\ (b_d/\tau)^2 \\ \sigma_c\ (3) &= 2\ (b_c/\tau)^2 \\ \hat{q}_d\ (3) &= \frac{1}{\tau}\ (\varepsilon_d^2 - b_d) \\ \hat{q}_c\ (3) &= \frac{1}{\tau}\ (\varepsilon_c^2 - b_c) \end{aligned}\right\} \quad \text{for } i=2 \text{ only}$$

$$\left.\begin{aligned} s_d &= e^{-\frac{\tau}{T}}[\sigma_d(i) - 2\ (b_d/\gamma)^2] + 2\ (b_d/\gamma)^2 \\ s_c &= e^{-\frac{\tau}{T}}[\sigma_c(i) - 2\ (b_c/\gamma)^2] + 2\ (b_c/\gamma)^2 \end{aligned}\right\}$$

$$\sigma_d(i+1) = \frac{2\, b_d^2\, s_d}{2\, b_d^2 + \tau^2 s_d}$$

$$\sigma_c(i+1) = \frac{2\, b_c^2\, s_c}{2\, b_c^2 + \tau^2 s_c}$$

$$\hat{q}_d(i+1) = \hat{q}_d(i) + \frac{s_d \tau}{2b_d^2 + s_d \tau^2} [\varepsilon_d^2 - b_d - r_d]$$

$$\hat{q}_c(i+1) = \hat{q}_c(i) + \frac{s_c \tau}{2b_c^2 + s_c \tau^2} [\varepsilon_c^2 - b_c - r_c]$$

for $i \geq 3$ only.

T is user-specified "forgetting time constant".

$$\alpha_d = \max \{0, \max \{\hat{q}_d(i+1), 0\} - \max \{\hat{q}_d(i), 0\}\}$$

$$\alpha_c = \max \{0, \max \{\hat{q}_c(i+1), 0\} - \max \{\hat{q}_c(i), 0\}\}$$

$$\sigma_d(i+1) = \sigma_d(i+1) + 2 \frac{\alpha_d}{\gamma} [2\, b_d + \alpha_d \tau]$$

$$\sigma_c(i+1) = \sigma_c(i+1) + 2 \frac{\alpha_c}{\gamma} [2\, b_c + \alpha_c \tau]$$

Step 4 (performed for $i \geq 2$ only) - Adjust Kalman filter update by successively computing

$$d_{11} = \alpha_d \cos^2\theta + \alpha_c \sin^2\theta$$

$$d_{12} = (\alpha_d - \alpha_c) \sin\theta \cos\theta$$

$$d_{22} = \alpha_d \sin^2\theta + \alpha_c \cos^2\theta$$

$$D = \left[\begin{array}{c|c} d_{11} & d_{12} \\ \hline d_{12} & d_{22} \end{array}\right]$$

(positive semi-definite 2x2 matrix)

$$P_3(t_{i+1}^+) = P_3(t_{i+1}^+) + \frac{1}{\gamma} D \qquad \text{(2x2 matrix)}$$

$$P_1(t_{i+1}^+) = P_1(t_{i+1}^+) + R(M_1 + R + D\tau)^{-1} D(M_1 + R + D\tau)^{-1} R\tau$$

(2x2 matrix)

$$\left.\begin{aligned} \hat{v}(t_{i+1}^+) &= \hat{v}(t_{i+1}^+) + \frac{\tau}{\gamma} D(M_1 + R + D\tau)(z - \bar{x}) \\ \hat{x}(t_{i+1}^+) &= \hat{x}(t_{i+1}^+) + R(M_1 + D\tau + R)D(M_1 + R)^{-1}(z - \bar{x})\tau \end{aligned}\right\} \text{2-vectors}$$

The adjustments in this step are explained in Ref. [3]. Corresponding adjustments for the maneuvering intensity estimation procedure constitute the last two computations listed here under Step 3.

## 5. ALTERNATE METHOD

This same kind of improvement can also be applied to the alternate method of Ref. [3] for adaptively estimating maneuvering intensity, in which the estimated maneuvering is contrained to be statistically isotropic, but which allows the inclusion of bearing - only input reports in the resulting tracking algorithm. In this case, only a single (non-negative) scalar maneuvering intensity parameter q is estimated, and the overall tracking algorithm can be summarized as follows for position reports:

Initialization - as in the algorithm of the preceding section, except that only a single value $\hat{q}(o)$ is specified, instead of $\hat{q}_d(o)$ and $\hat{q}_c(o)$.

From $t_i^+$ to $t_{i+1}^+$ ; i = 0, 1, ...:

Step 1 - calculate the 2x2 matrix Q by the steps

$$q = \max\{\hat{q}(i), 0\}$$

$$Q = \left[\begin{array}{c|c} q & 0 \\ \hline 0 & q \end{array}\right],$$

Then continue as in the preceding section.

Step 2 - as in the preceding section.

Step 3 - For i=0,1, just set $\hat{q}(i+1) = \hat{q}(i)$.
Otherwise, compute in turn:

$$\left[\begin{array}{c|c} g_{11} & g_{12} \\ \hline g_{12} & g_{22} \end{array}\right] = A + R \qquad \text{(2x2 matrix)}$$

$$a^2 = g_{11} + g_{22}$$

$$b^2 = \sqrt{(g_{11} - g_{22})^2 + 4\, g_{12}^2}$$

$$a^2 = \frac{1}{2}\,(a^2 + b^2)$$

$$b^2 = a^2 - b^2$$

$$\cos \beta = \frac{a^2 - g_{11}}{\sqrt{(a^2 - g_{11})^2 + g_{12}^2}}$$

$$\sin \beta = \frac{g_{12}}{\sqrt{(a^2 - g_{11})^2 + g_{12}^2}}$$

Note: a and b are the semi-major and semi-minor axes of the sigma-ellipse for the bivariate Normal distribution with covariance matrix A + R; $\beta$ is the angle between the semi-major axis and the 1-coordinate axis.

$$\left[\begin{array}{c} \varepsilon_a \\ \hline \varepsilon_b \end{array}\right] = \left[\begin{array}{c|c} \cos\beta & \sin\beta \\ \hline -\sin\beta & \cos\beta \end{array}\right] (z-\bar{x}) \qquad \text{(2-vector)} \qquad (21)$$

$$h = g_{11} \sin^2 \beta - 2g_{12} \sin \beta \cos \beta + g_{22} \cos^2 \beta + q\tau$$

$$\eta = b^2 + q\tau$$

$$k = a^2 + q\tau \qquad (22)$$

$$\left.\begin{aligned} \sigma(3) &= 2\left(\frac{h}{\tau}\right)^2 \\ \hat{q}(3) &= \frac{1}{\tau}(\varepsilon_b^2 - \eta) \end{aligned}\right\} \text{ for } i=2 \text{ only}$$

$$\left.\begin{aligned} s &= e^{-\frac{\tau}{T}} \left[\sigma(i) - 2\left(\frac{h}{\gamma}\right)^2\right] + 2\left(\frac{h}{\gamma}\right)^2 \\ \sigma(i+1) &= \frac{2h^2 s}{2h^2 + s\tau} \\ \hat{q}(i+1) &= \hat{q}(i) + \frac{s\tau}{2h^2 + s\tau^2}\left[\varepsilon_b^2 - \eta\right] \end{aligned}\right\} \text{ for } i \geq 3 \text{ only}$$

$$\sigma(i+1) = \frac{2k^2}{2k^2 + s\tau}\,\sigma(i+1) \qquad (23)$$

$$\hat{q}(i+1) = \hat{q}(i+1) + \frac{s\tau}{2k^2 + s\tau^2}\left[\varepsilon_a^2 - k\right] \qquad (24)$$

$$\alpha = \max\,\{0, \max\,\{\hat{q}(i+1),0\} - \max\,\{\hat{q}(i),0\}\}$$

$$\sigma(i+1) = \sigma(i+1) + 2\,\frac{\alpha}{\gamma}\,[2h + \alpha\tau]$$

Step 4 - calculate the 2x2 matrix D as $D = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix}$ ,

then continue as in the preceding section.

The advantage of this alternate method is that it can make effective use of bearing-only reports of ship location (one of which specifies a time, an observed bearing from a specified point at that time, and a bearing variance), or what is almost the same thing, position reports with long, narrow error ellipses. One simple way of doing this with such a bearing report is to replace it with an approximately equivalent position report, which would have a large variance component along the line-of-bearing, and use the procedure as just described. The best way of constructing an equivalent position report depends on the bearing sensor range, how much the projected ship position distribution at the time of the report "overlaps" the bearing sensor location, and the distance of the estimated ship position at that time from the sensor location. The main idea is to have an appropriate containment ellipse for this (bivariate Normally distributed) position report, say the two-sigma (86%) ellipse, match the corresponding wedge-shaped confidence region of the bearing report as well as possible within the range of likely projected ship locations. As a numerically more efficient alternative when the predicted ship range from the bearing sensor is large compared to the uncertainty in this prediction, one could also use a bearing-only report as a scalar measurement of the cross-bearing component of ship position with variance $r^2\delta^2$, where r is the projected ship range (from the sensor) and $\delta^2$ is the bearing variance in radians. This would require the replacement of Step 2 (in this section) by the appropriate Kalman filter updating procedure for such a single-component measurement. It is also necessary to alter Step 3 by computing $b^2$ as $r^2\delta^2$ plus the cross-bearing component of $M_1$, computing $\varepsilon_b$ as the cross-bearing component of the observed minus predicted ship position (as evaluated from the observed bearing at the predicted ship range) in lieu of Eq. (21), and skipping the computations of Eqs. (22)-(24) and those prior to Eq. (21).

For position reports which are well-localized in both dimensions, the use of this alternate tracking algorithm would probably not cause much loss of precision over the algorithm of the preceding section. It was originally thought that allowing the estimates of the down-track and cross-track components of the maneuvering intensiity to be different, as they may be in the algorithm of the preceding section, would often lead to more precise estimates of ship motion when a ship zigzags about an average course, or changes speed a lot but stays on the same heading. In practice, however, the estimated maneuvering is usually fairly isotropic anyway.

## 6. DISCUSSION

To summarize the basic usage of the two ship tracking algorithm variants presented here for the case of position reports, the user specifies the two (scalar) parameters:

$\lambda$ = Prior estimate of ship speed (i.e., average speed for anticipated target population). Making $\lambda$ too large doesn't matter much except when this algorithm is used as part of a multitarget tracker, but making $\lambda$ too small does.

T = Average length of time over which a ship's maneuvering behavior is judged to remain statistically the same. This value can often be made infinite without serious consequences; another reasonable possibility is to set $T = \sqrt{A/\lambda}$, where A is the area of the surveillance region in which ships are being tracked.

Also, it is sometimes slightly advantageous to specify some nonzero initial value for $\hat{q}_d(0)$ and $\hat{q}_c(0)$, or $\hat{q}(0)$ in the case of the variant of Section 5. For either variant, the algorithm then operates on input consisting of a time-ordered sequence of ship position reports, each of which contains the following six data:

1. time of report

2-3. two coordinates of observed ship position at that time

4-6. the three independent components of the (symmetric 2x2) covariance matrix for the error in this observed position.

As output, it provides (a) estimates of ship position and average velocity, at present or future times, (b) error covariance matrices (4x4) that correspond to these estimates, and (c) estimates and corresponding error covariances of maneuverability parameters (two in the variant of Section 4, one in the variant of Section 5).

Normally, the only outputs of interest are the position components of the estimated position-velocity state vector, and the corresponding 2x2 error covariance matrix. It is also usual to regard these errors and the position observation errors in the input reports as bivariate Normally distributed, and to use equivalent parameters specifying the two-sigma ellipses (86% containment ellipses) of these distributions in place of their covariance matrices.

The ellipse parameters normally used for this purpose are the semi-major and semi-minor axis lengths and the angle of the major axis measured clockwise from local north. As examples of what this all amounts to in practice, experimental FORTRAN implementations of both variants of this overall ship tracking algorithm are listed in the appendix for the usual case in which the input and output error distributions are specified in terms of containment ellipse parameters instead of covariance matrix components. The input and output are from a terminal in these implementations.

Figures 2 and 3 show an example of the comparative performances of this tracking algorithm (the version of Section 4 here) and the corresponding one of Ref. [3], which uses the former method of adaptive maneuvering estimation. The parameter values $\hat{q}_d(0) = \hat{q}_c(0) = 0$ and $T = \infty$ were used in each case. For clarity in these figures, the two-sigma (86% containment) ellipses for the output location estimates are displayed only for selected times. The new algorithm is not shown to its best advantage in this example because the position reports are all evenly spaced in time. Even so, however, the accuracy of the estimated ship positions is as good as that of the former algorithm, and the containment ellipses generated by the new algorithm are generally somewhat larger, which is an indication of better statistical consistency in light of the computational experience with actual tracking data mentioned earlier.

The variant of Section 5 here has also been embedded in a multi-target tracker, in the manner described in Ref. [4], as a replacement for the corresponding single-target tracking method of Ref. [3], which estimates only a scalar intensity for statistically isotropic maneuvering. The input data in this application were mostly bearing-only reports which occurred in sporadic bursts for each ship, and thus were very unevenly spaced in time. In this case there was a substantial improvement in performance with the tracker incorporating the new adaptive maneuvering estimation procedure described in this report.

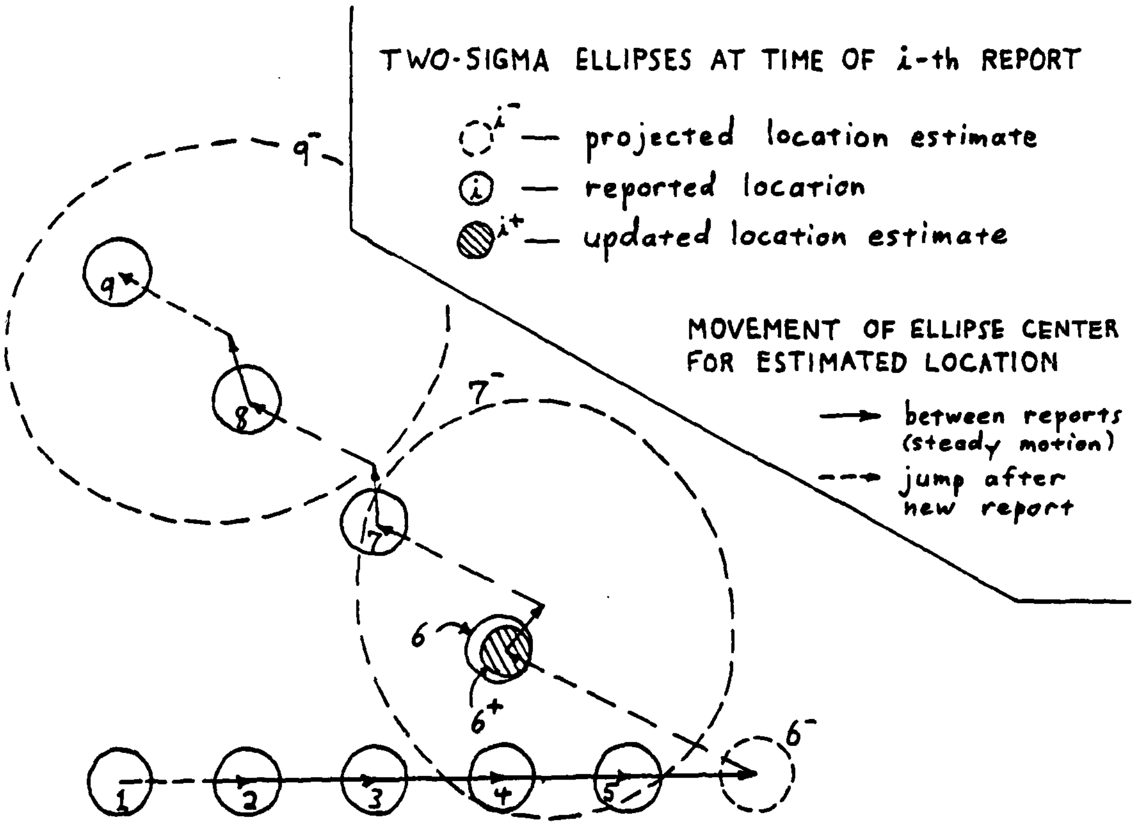



Fig. 2 — Tracking performance with former procedure

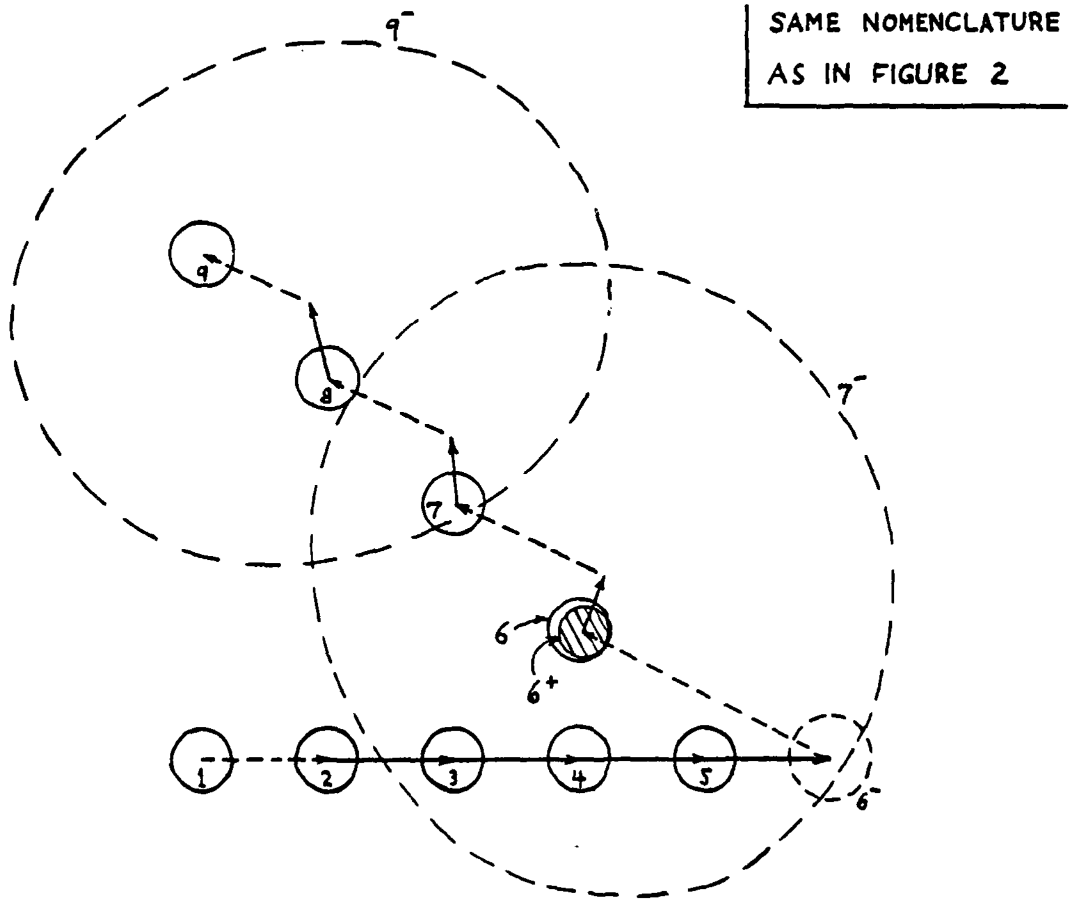



Fig. 3 — Tracking performance with procedure of Section 4

## 7. REFERENCES

[1] W.W. Willman, "Recursive Filtering Algorithms for Ship Tracking," NRL Report 7969, April 6, 1976.

[2] W.W. Willman, "Some Refinements for a Ship Tracking Algorithm," NRL Memorandum Report 3991, May 2, 1979.

[3] W.W. Willman, "Modifications of a Ship Tracking Algorithm for Maneuver Following and Bearing-Only Data," NRL Memorandum Report 4579, October 1, 1981.

[4] W.W. Willman, "Some Performance Results for Recursive Multitarget Correlator-Tracker Algorithms," NRL Report 8423, July 23, 1980.

## APPENDIX - PROGRAM LISTINGS

```
C          KALMAN FILTER WITH ADAPTIVE DRIVING NOISE
C
C  FOR TRACKING IN RECTANGULAR X,Y-COORDINATES WITH POSITION
C  REPORTS WHICH ARE MEANINGFULLY LOCALIZED IN TWO DIMENSIONS.
C
C
C                INITIALIZATION
C
C
C   ENTER USER-SPECIFIED PARAMETER VALUES FROM TERMINAL
C
      ACCEPT"PRIOR EXPECTED SHIP SPEED = ",B
C
C   COMPUTE INITIAL VALUES
C
      HI=0.
      HC=0.
      CI=0.
      CC=0.
      U=0.
      V=0.
      PXU=0.
      PXV=0.
      PYU=0.
      PYV=0.
      PUU=.5*B*B
      PUV=0.
      PVV=PUU
      QXX=0.
      QXY=0.
      QYY=0.
C
C
C                BEGIN TRACKING
C
C
C   ENTER AND PROCESS INITIAL REPORT
C
C  IR = INDICATOR: ZERO IF "REPORT" JUST SPECIFIES A TIME AT
C                  WHICH DEAD-RECKONED OUTPUT IS DESIRED,
C                  POSITIVE IF IT IS AN ACTUAL REPORT OF
C                  OBSERVED POSITION, NEGATIVE TO
C                  TERMINATE TRACKING.
C
C             FOR POSITIVE IR:
C  T = TIME.
C  ZX,ZY = OBSERVED POSITION IN X,Y-COORDINATES.
C  SMA = SEMIMAJOR AXIS OF 86% CONTAINMENT ELLIPSE FOR
C        OBSERVATION.
C  SMI = SEMIMINOR AXIS OF CONTAINMENT ELLIPSE.
C  THT = ORIENTATION OF MAJOR AXIS (DEGREES CLOCKWISE FROM
C        Y-AXIS).
C
      ACCEPT"IR,T,ZX,ZY,SMA,SMI,THT = ",IR,T,ZX,ZY,SMA,SMI,THT
      IF(IR.LE.0) GO TO 99
      I=0
      THT=THT/57.3
      X=ZX
      Y=ZY
      ST=SIN(THT)
      CT=COS(THT)
      SMA=SMA*SMA
      SMI=SMI*SMI
```

```
      PXX=.25*(SMA*ST*ST+SMI*CT*CT)
      PXY=.25*ST*CT*(SMA-SMI)
      PYY=.25*(SMA*CT*CT+SMI*ST*ST)
      TN=T
      TL=T
      TLU=T
      GO TO 40
C
C
C               CONTINUE TRACKING
C
C
C    ENTER NEXT REPORT
C
   90 ACCEPT"IR,T,ZX,ZY,SMA,SMI,THT = ",IR,T,ZX,ZY,SMA,SMI,THT
      IF(IR.LT.0) GO TO 99
      IF(IR.EQ.0) GO TO 10
      THT=THT/57.3
      ST=SIN(THT)
      CT=COS(THT)
      SMA=SMA*SMA
      SMI=SMI*SMI
      RXX=.25*(SMA*ST*ST+SMI*CT*CT)
      RXY=.25*ST*CT*(SMA-SMI)
      RYY=.25*(SMA*CT*CT+SMI*ST*ST)
C
C    PROJECT STATE DENSITY TO CURRENT TIME
C
   10 TAU=T-TL
      TL=T
      XB=X+U*TAU
      YB=Y+V*TAU
      GXX=PXX+(2.*PXU+QXX)*TAU+PUU*TAU*TAU
      GXY=PXY+(PXV+PYU+QXY)*TAU+PUV*TAU*TAU
      GYY=PYY+(2.*PYV+QYY)*TAU+PVV*TAU*TAU
      GXU=PXU+PUU*TAU
      GXV=PXV+PUV*TAU
      GYU=PYU+PUV*TAU
      GYV=PYV+PVV*TAU
      IF(IR.NE.0) GO TO 60
      X=XB
      Y=YB
      PXX=GXX
      PXY=GXY
      PYY=GYY
      PXU=GXU
      PXV=GXV
      PYU=GYU
      PYV=GYV
      GO TO 40
C
C    UPDATE STATE DENSITY WITH NEW REPORT
C
   60 TAU=T-TLU
      TLU=T
      EX=ZX-XB
      EY=ZY-YB
      G1=GXX+RXX
      G2=GXY+RXY
      G3=GYY+RYY
      DET=G1*G3-G2*G2
      DETI=0.
      IF(DET.GT.0.) DETI=1./DET
```

```
      HXX=G3*DETI
      HXY=-G2*DETI
      HYY=G1*DETI
      X=XB+(GXX*HXX+GXY*HXY)*EX+(GXX*HXY+GXY*HYY)*EY
      Y=YB+(GXY*HXX+GYY*HXY)*EX+(GXY*HXY+GYY*HYY)*EY
      U=U+(GXU*HXX+GYU*HXY)*EX+(GXU*HXY+GYU*HYY)*EY
      V=V+(GXV*HXX+GYV*HXY)*EX+(GXV*HXY+GYV*HYY)*EY
      PXX=GXX-GXX*GXX*HXX-2.*GXX*GXY*HXY-GXY*GXY*HYY
      PXY=GXY-GXX*GXY*HXX-(GXX*GYY+GXY*GXY)*HXY-GXY*GYY*HYY
      PYY=GYY-GXY*GXY*HXX-2.*GXY*GYY*HXY-GYY*GYY*HYY
      PXU=GXU-GXX*GXU*HXX-(GXX*GYU+GXY*GXU)*HXY-GXY*GYU*HYY
      PXV=GXV-GXX*GXV*HXX-(GXX*GYV+GXY*GXV)*HXY-GXY*GYV*HYY
      PYU=GYU-GXY*GXU*HXX-(GXY*GYU+GYY*GXU)*HXY-GYY*GYU*HYY
      PYV=GYV-GXY*GXV*HXX-(GXY*GYV+GYY*GXV)*HXY-GYY*GYV*HYY
      PUU=PUU-GXU*GXU*HXX-2.*GXU*GYU*HXY-GYU*GYU*HYY
      PUV=PUV-GXU*GXV*HXX-(GXU*GYV+GYU*GXV)*HXY-GYU*GYV*HYY
      PVV=PVV-GXV*GXV*HXX-2.*GXV*GYV*HXY-GYV*GYV*HYY
C
C   UPDATE DRIVING NOISE ESTIMATE AND ADJUST ESTIMATOR PARAMETERS
C
      IF(TAU.LE.0.) GO TO 40
      TE=T-IN
      IF(TE.LE.0.) GO TO 40
      I=I+1
      IF(I.EQ.1) GO TO 40
      U2=U*U
      V2=V*V
      SQ=U2+V2
      QINV=0.
      IF(SQ.GT.0.) QINV=1./SQ
      U2=U2*QINV
      V2=V2*QINV
      UV=U*V*QINV
      SIT=QINV*(U*EX+V*EY)**2
      SCT=QINV*(U*EY-V*EX)**2
      GIT=U2*GXX+2.*UV*GXY+V2*GYY
      GCT=V2*GXX-2.*UV*GXY+U2*GYY
      BIT=U2*G1+2.*UV*G2+V2*G3
      BCT=V2*G1-2.*UV*G2+U2*G3
      IF(I.GT.2) GO TO 30
      GNI=1./TAU
      GNC=GNI
      VRI=2.*GIT*GIT*GNI*GNI
      VRC=2.*GCT*GCT*GNC*GNC
      GO TO 50
   30 DEN=2.*GIT*GIT+VRI*TAU*TAU
      DENI=0.
      IF(DEN.GT.0.) DENI=1./DEN
      GNI=VRI*TAU*DENI
      VRI=VRI*(1.-VRI*TAU*TAU*DENI)
      DEN=2.*GCT*GCT+VRC*TAU*TAU
      DENI=0.
      IF(DEN.GT.0.) DENI=1./DEN
      GNC=VRC*TAU*DENI
      VRC=VRC*(1.-VRC*TAU*TAU*DENI)
   50 CI=CI+GNI*(SIT-BIT)
      CC=CC+GNC*(SCT-BCT)
      OI=HI
      OC=HC
      HI=0.
      HC=0.
      IF(CI.GT.0.) HI=CI
      IF(CC.GT.0.) HC=CC
```

```
      DI=HI-DI
      DC=HC-DC
      QXX=U2*HI+V2*HC
      QXY=UV*(HI-HC)
      QYY=V2*HI+U2*HC
      PI=0.
      PC=0.
      IF(DI.GT.0.) PI=DI/TE
      IF(DC.GT.0.) PC=DC/TE
      VRI=VRI+2.*PI*(PI*TE*TAU+2.*GIT)
      VRC=VRC+2.*PC*(PC*TE*TAU+2.*GCT)
C
C   ADJUST PARAMETERS OF STATE DENSITY
C
      EZ=HXX*EX+HXY*EY
      EW=HXY*EX+HYY*EY
      HXX=U2*PI+V2*PC
      HXY=UV*(PI-PC)
      HYY=V2*PI+U2*PC
      PUU=PUU+HXX
      PUV=PUV+HXY
      PVV=PVV+HYY
      AXX=HXX*TAU
      AXY=HXY*TAU
      AYY=HYY*TAU
      GXX=AXX*TE
      GXY=AXY*TE
      GYY=AYY*TE
      G1=G1+GXX
      G2=G2+GXY
      G3=G3+GYY
      DET=G1*G3-G2*G2
      DETI=0.
      IF(DET.GT.0.) DETI=1./DET
      HXX=G3*DETI
      HXY=-G2*DETI
      HYY=G1*DETI
      U=U+(AXX*HXX+AXY*HXY)*EX+(AXX*HXY+AXY*HYY)*EY
      V=V+(AXY*HXX+AYY*HXY)*EX+(AXY*HXY+AYY*HYY)*EY
      G1=RXX*HXX+RXY*HXY
      G2=RXX*HXY+RXY*HYY
      G3=RXY*HXX+RYY*HXY
      G4=RXY*HXY+RYY*HYY
      PXX=PXX+GXX*G1*G1+2.*GXY*G1*G2+GYY*G2*G2
      PXY=PXY+GXX*G1*G3+GXY*(G1*G4+G2*G3)+GYY*G2*G4
      PYY=PYY+GXX*G3*G3+2.*GXY*G3*G4+GYY*G4*G4
      X=X+GXX*G1*EZ+GXY*(G1*EW+G2*EZ)+GYY*G2*EW
      Y=Y+GXX*G3*EZ+GXY*(G3*EW+G4*EZ)+GYY*G4*EW
C
C
C              OUTPUT
C
C
   40 C1=PXX+PYY
      C2=SQRT((PXX-PYY)**2+4.*PXY*PXY)
      C1=.5*(C1+C2)
      C2=C1-C2
      SMA=2.*SQRT(C1)
      SMI=2.*SQRT(C2)
      IF(PXY.NE.0.) GO TO 70
      THT=0.
```

```
      IF(PXX.GT.PYY) THT=90.
      GO TO 80
   70 THT=57.3*ATAN((PXX-C1)/PYY)+90.
C
C   DISPLAY OUTPUT ON TERMINAL
C
C  IR = ABOVE-DESCRIBED INDICATOR FOR "REPORT" JUST PROCESSED.
C  T = CURRENT TIME (= TIME OF THIS REPORT).
C  X,Y = CURRENT ESTIMATED POSITION IN X,Y-COORDINATES.
C  SMA = SEMIMAJOR AXIS OF 86% CONTAINMENT ELLIPSE FOR
C        CURRENT POSITION ESTIMATE.
C  SMI = SEMIMINOR AXIS OF CONTAINMENT ELLIPSE.
C  THT = ORIENTATION OF MAJOR AXIS (DEGREES CLOCKWISE
C        FROM Y-AXIS.
C
   80 TYPE" "
      TYPE"IR = ",IR
      TYPE"T = ",T
      TYPE"X,Y = ",X,Y
      TYPE"SMA,SMI = ",SMA,SMI
      TYPE"THT = ",THT
      TYPE" "
C
C
C          PROCESS NEXT REPORT (OR DEAD-RECKON TIME), IF ANY
C
C
      GO TO 90
C
C
C                TERMINATION
C
C
   99 CONTINUE
      END
```

```
C          KALMAN FILTER WITH ADAPTIVE DRIVING NOISE
C
C  FOR TRACKING IN RECTANGULAR X,Y-COORDINATES WITH POSITION
C  REPORTS WHICH ARE MEANINGFULLY LOCALIZED IN ONLY ONE
C  DIMENSION, SUCH AS APPROXIMATIONS OF BEARING-ONLY REPORTS.
C
C
C                 INITIALIZATION
C
C
C   ENTER USER-SPECIFIED PARAMETER VALUES FROM TERMINAL
C
      ACCEPT"PRIOR EXPECTED SHIP SPEED = ",B
C
C   COMPUTE INITIAL VALUES
C
      CI=0.
      U=0.
      V=0.
      PXU=0.
      PXV=0.
      PYU=0.
      PYV=0.
      PUU=.5*B*B
      PUV=0.
      PVV=PUU
      Q=0.
C
```

```
C
C                BEGIN TRACKING
C
C
C   ENTER AND PROCESS INITIAL REPORT
C
C  IR = INDICATOR: ZERO IF "REPORT" JUST SPECIFIES A TIME AT
C                  WHICH DEAD-RECKONED OUTPUT IS DESIRED,
C                  POSITIVE IF IT IS AN ACTUAL REPORT OF
C                  OBSERVED POSITION, NEGATIVE TO
C                  TERMINATE TRACKING.
C
C              FOR POSITIVE IR:
C  T = TIME.
C  ZX,ZY = OBSERVED POSITION IN X,Y-COORDINATES.
C  SMA = SEMIMAJOR AXIS OF 86% CONTAINMENT ELLIPSE FOR
C        OBSERVATION.
C  SMI = SEMIMINOR AXIS OF CONTAINMENT ELLIPSE.
C  THT = ORIENTATION OF MAJOR AXIS (DEGREES CLOCKWISE FROM
C        Y-AXIS).
C
      ACCEPT"IR,T,ZX,ZY,SMA,SMI,THT = ",IR,T,ZX,ZY,SMA,SMI,THT
      IF(IR.LE.0) GO TO 99
      I=0
      THT=THT/57.3
      X=ZX
      Y=ZY
      ST=SIN(THT)
      CT=COS(THT)
      SMA=SMA*SMA
      SMI=SMI*SMI
      PXX=.25*(SMA*ST*ST+SMI*CT*CT)
      PXY=.25*ST*CT*(SMA-SMI)
      PYY=.25*(SMA*CT*CT+SMI*ST*ST)
      TN=T
      TL=T
      TLU=T
      GO TO 40
C
C
C                CONTINUE TRACKING
C
C
C   ENTER NEXT REPORT
C
   90 ACCEPT"IR,T,ZX,ZY,SMA,SMI,THT = ",IR,T,ZX,ZY,SMA,SMI,THT
      IF(IR.LT.0) GO TO 99
      IF(IR.EQ.0) GO TO 10
      THT=THT/57.3
      ST=SIN(THT)
      CT=COS(THT)
      SMA=SMA*SMA
      SMI=SMI*SMI
      RXX=.25*(SMA*ST*ST+SMI*CT*CT)
      RXY=.25*ST*CT*(SMA-SMI)
      RYY=.25*(SMA*CT*CT+SMI*ST*ST)
C
C   PROJECT STATE DENSITY TO CURRENT TIME
C
   10 TAU=T-TL
      TL=T
      XB=X+U*TAU
      YB=Y+V*TAU
```

```
      GA=PXX+2.*PXU*TAU+PUU*TAU*TAU
      GB=PYY+2.*PYV*TAU+PVV*TAU*TAU
      GXX=GA+Q*TAU
      GXY=PXY+(PXV+PYU)*TAU+PUV*TAU*TAU
      GYY=GB+Q*TAU
      GXU=PXU+PUU*TAU
      GXV=PXV+PUV*TAU
      GYU=PYU+PUV*TAU
      GYV=PYV+PVV*TAU
      IF(IR.NE.0) GO TO 60
      X=XB
      Y=YB
      PXX=GXX
      PXY=GXY
      PYY=GYY
      PXU=GXU
      PXV=GXV
      PYU=GYU
      PYV=GYV
      GO TO 40
C
C    UPDATE STATE DENSITY WITH NEW REPORT
C
   60 TAU=T-TLU
      TLU=T
      EX=ZX-XB
      EY=ZY-YB
      G1=GXX+RXX
      G2=GXY+RXY
      G3=GYY+RYY
      DET=G1*G3-G2*G2
      DETI=0.
      IF(DET.GT.0.) DETI=1./DET
      HXX=G3*DETI
      HXY=-G2*DETI
      HYY=G1*DETI
      X=XB+(GXX*HXX+GXY*HXY)*EX+(GXX*HXY+GXY*HYY)*EY
      Y=YB+(GXY*HXX+GYY*HXY)*EX+(GXY*HXY+GYY*HYY)*EY
      U=U+(GXU*HXX+GYU*HXY)*EX+(GXU*HXY+GYU*HYY)*EY
      V=V+(GXV*HXX+GYV*HXY)*EX+(GXV*HXY+GYV*HYY)*EY
      PXX=GXY-GXX*GXX*HXX-2.*GXX*GXY*HXY-GXY*GXY*HYY
      PXY=GXY-GXX*GXY*HXX-(GXX*GYY+GXY*GXY)*HXY-GXY*GYY*HYY
      PYY=GYY-GXY*GXY*HXX-2.*GXY*GYY*HXY-GYY*GYY*HYY
      PXU=GXU-GXX*GXU*HXX-(GXX*GYU+GXY*GXU)*HXY-GXY*GYU*HYY
      PXV=GXV-GXX*GXV*HXX-(GXX*GYV+GXY*GXV)*HXY-GXY*GYV*HYY
      PYU=GYU-GXY*GXU*HXX-(GXY*GYU+GYY*GXU)*HXY-GYY*GYU*HYY
      PYV=GYV-GXY*GXV*HXX-(GXY*GYV+GYY*GXV)*HXY-GYY*GYV*HYY
      PUU=PUU-GXU*GXU*HXX-2.*GXU*GYU*HXY-GYU*GYU*HYY
      PUV=PUV-GXU*GXV*HXX-(GXU*GYV+GYU*GXV)*HXY-GYU*GYV*HYY
      PVV=PVV-GXV*GXV*HXX-2.*GXV*GYV*HXY-GYV*GYV*HYY
C
C    UPDATE DRIVING NOISE ESTIMATE AND ADJUST ESTIMATOR PARAMETERS
C
      IF(TAU.LE.0.) GO TO 40
      TE=T-TN
      IF(TE.LE.0.) GO TO 40
      I=I+1
      IF(I.EQ.1) GO TO 40
      DI=I
      G1=GA+RXX
      G3=GB+RYY
      C1=G1+G3
      C2=SQRT((G1-G3)**2+4.*G2*G2)
```

```
      C1=.5*(C1+C2)
      C2=C1-C2
      CB=C1-G1
      DEN=CB*CB+G2*G2
      DENI=0.
      IF(DEN.GT.0.) DENI=1./DEN
      IF(C2.GT..99*C1) CB=1.
      IF(C2.GT..99*C1) DENI=1.
      EA=DENI*(CB*EX+G2*EY)**2
      EB=DENI*(CB*EY-G2*EX)**2
      H=(G2*G2*GXX-2.*G2*CB*GXY+CB*CB*GYY)*DENI
      HB=C2*Q*TAU
      IF(I.GT.2) GO TO 30
      GNI=1./TAU
      VRI=2.*H*H*GNI*GNI
      GO TO 50
   30 DEN=2.*H*H+VRI*TAU*TAU
      DENI=0.
      IF(DEN.GT.0.) DENI=1./DEN
      GNI=VRI*TAU*DENI
      VRI=VRI*(1.-VRI*TAU*TAU*DENI)
   50 CI=CI+GNI*(EB-HB)
      Q=0.
      IF(CI.GT.0.) Q=CI
      HA=C1*Q*TAU
      DEN=2.*HA*HA+VRI*TAU*TAU
      DENI=0.
      IF(DEN.GT.0.) DENI=1./DEN
      GNI=VRI*TAU*DENI
      VRI=VRI*(1.-VRI*TAU*TAU*DENI)
      CI=CI+GNI*(EA-HA)
      Q=0.
      IF(CI.GT.0.) Q=CI
      DI=Q-DI
      PI=0.
      IF(DI.GT.0.) PI=DI/TE
      VRI=VRI+2.*PI*(PI*TE*TAU+2.*H)
C
C   ADJUST PARAMETERS OF STATE DENSITY
C
      EZ=HXX*EX+HXY*EY
      EW=HXY*EX+HYY*EY
      PUU=PUU+PI
      PVV=PVV+PI
      A=PI*TAU
      G=A*TE
      G1=G1+G
      G3=G3+G
      DET=G1*G3-G2*G2
      DETI=0.
      IF(DET.GT.0.) DETI=1./DET
      HXX=G3*DETI
      HXY=-G2*DETI
      HYY=G1*DETI
      U=U+A*(HXX*EX+HXY*EY)
      V=V+A*(HXY*EX+HYY*EY)
      G1=HXX*HXX+HXY*HXY
      G2=HXX*HXY+HXY*HYY
      G3=HXY*HXX+HYY*HXY
      G4=HXY*HXY+HYY*HYY
      PXX=PXX+G*(G1*G1+G2*G2)
      PXY=PXY+G*(G1*G3+G2*G4)
      PYY=PYY+G*(G3*G3+G4*G4)
```

```
      X=X+G*(G1*EZ+G2*EW)
      Y=Y+G*(G3*EZ+G4*EW)
C
C
C                 OUTPUT
C
C
   40 C1=PXX+PYY
      C2=SQRT((PXX-PYY)**2+4.*PXY*PXY)
      C1=.5*(C1+C2)
      C2=C1-C2
      SMA=2.*SQRT(C1)
      SMI=2.*SQRT(C2)
      IF(PXY NE.0.) GO TO 70
      THT=0.
      IF(PXX.GT.PYY) THT=90.
      GO TO 80
   70 THT=57.3*ATAN((PXX-C1)/PXY)+90.
C
C   DISPLAY OUTPUT ON TERMINAL
C
C  IR = ABOVE-DESCRIBED INDICATOR FOR "REPORT" JUST PROCESSED.
C  T = CURRENT TIME (= TIME OF THIS REPORT).
C  X,Y = CURRENT ESTIMATED POSITION IN X,Y-COORDINATES.
C  SMA = SEMIMAJOR AXIS OF 86% CONTAINMENT ELLIPSE FOR
C          CURRENT POSITION ESTIMATE.
C  SMI = SEMIMINOR AXIS OF CONTAINMENT ELLIPSE.
C  THT = ORIENTATION OF MAJOR AXIS (DEGREES CLOCKWISE
C          FROM Y-AXIS.
C
   80 TYPE" "
      TYPE"IR = ",IR
      TYPE"T = ",T
      TYPE"X,Y = ",X,Y
      TYPE"SMA,SMI = ",SMA,SMI
      TYPE"THT = ",THT
      TYPE" "
C
C
C            PROCESS NEXT REPORT (OR DEAD-RECKON TIME), IF ANY
C
C
      GO TO 90
C
C
C                 TERMINATION
C
C
   99 CONTINUE
      END
```

END
DATE
FILMED
5-82
DTIC