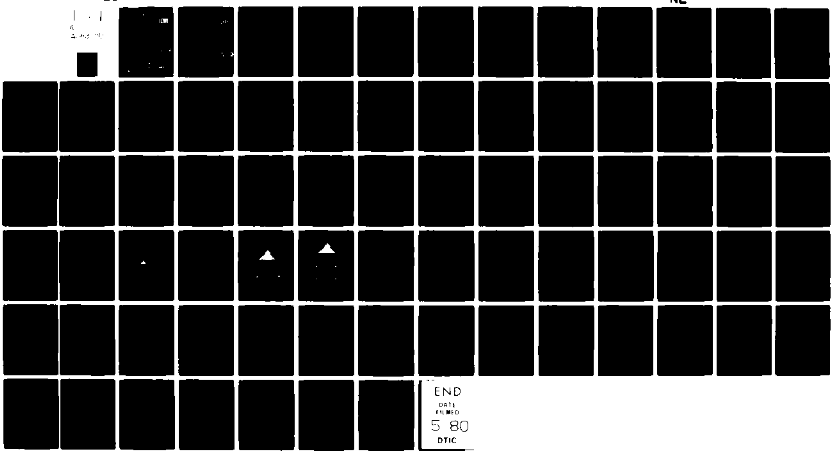
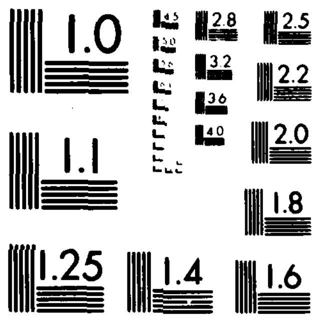
AD-A083 220 ORINCON CORP LA JOLLA CA F/G 17/7
A MULTI-TARGET SURVEY. VOLUME I.(U)
DEC 79 N00014-77-C-0296
UNCLASSIFIED OC-R-79-0296-1-VOL-1 NL




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

(1)
b.s.

ORINCON

ADA083220

DDC FILE COPY

DTIC
S ELECTE D
APR 2 1 1980
B

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

80 4 17 163

Operations Research INformation CONtrol

ORINCON Corporation

3366 No. Torrey Pines Ct., Suite 320, La Jolla, CA 92037 (714) 455-5530

14 OC-R-79-0296-1-Vol-1

11 December 1979

12 15

1 LEVEL II

Volume I

9 FINAL REPORT

6 A MULTI-TARGET SURVEY. Vol. I.

Contract ~~No.~~ N00014-77-C-0296 15

Approved for public release;
distribution unlimited

DTIC
ELECTE
S APR 21 1980 D
B

Submitted to:

Office of Naval Research
Department of the Navy - Code 613B:NCM
800 N. Quincy Street
Arlington, Virginia 22217

80 1 21 002

Operations Research INformation CONtrol

TABLE OF CONTENTS

NOTE:
Volume II of this Final Report is bound separately and contains publications credited to ONR Contract No. N00014-77-C-0296.

SECTION I

INTRODUCTION

The work done under Contract No. N00014-77-C-0296 is described in a number of different places which are drawn together and summarized in this final report. First, the publications credited to this contract which are available from the open literature are listed below.

1. "A Score for Correct Data Association in Multi-Target Tracking," by D. L. Alspach and R. N. Lobbia, appears in the December 1979 Proceedings of the Decision and Control Conference held in Fort Lauderdale, Florida.

2. "Sound Speed Estimation as a Means of Improving Target Tracking Performance," by D. L. Alspach, G. L. Mohnkern, and R. N. Lobbia, appears in the Proceedings of the 13th Asilomar Conference on Circuits, Systems, and Computers, November 1979, Pacific Grove, California.

3. "Multiple Coherence," by R. D. Trueblood and D. L. Alspach, was presented and appears in the Proceedings of the November 1978 Conference on Systems Science.

4. "A Case Study in Adaptive Sound Speed Estimation," by R. N. Lobbia and D. L. Alspach, was presented at and appears in the Proceedings of the November 1978 Asilomar Conference on Systems Science.

5. "Data Association Algorithms for Large Area Surveillance," by C. M. Petersen and C. L. Morefield, ORSA/TIMS Meeting, May 1978.

6. "Application of 0-1 Integer Programs to Multi-Target Tracking Problems," by C. L. Morefield, IEEE Transactions on Automatic Control, June 1977.

Copies of the papers published during the second year of the contract are included as Volume II of this report. Additional copies of all the above have been transmitted to ONR.

The remainder of the work performed under this contract to ONR is of two different types. A specific approach to reducing the computational problem in implementing a certain class of multi-target problems is contained in Section II. This section has been submitted in essentially this form to Information Sciences for publication.

The second part of the work is a general discussion of all aspects of the overall ocean monitoring and control problem. This essentially independent white paper is entitled "A Discussion of the Ocean Tactical Targeting Problem" and is included as Section III of this final report.

SECTION II

AN EFFICIENT TRACK ASSOCIATION ALGORITHM FOR THE MULTITARGET TRACKING PROBLEM

ABSTRACT

As the final contribution for this contract, a new and extremely promising approach is presented for the solution of the multitarget tracking problem for an important class of targets. For the class of targets that can be described as a linear, time-invariant system, it appears that this approach may be considerably more efficient, computationally, than existing procedures.

A survey of approaches to the solution of the multitarget tracking problem has been presented in Reference 2. As indicated there, most algorithms require the use of the Kalman filter to provide state estimation for data association and track assignment. In this report, the need for state estimation to identify feasible tracks is eliminated. The use of simpler input-output models for track prediction results in a very substantial reduction in computational burden when compared with methods requiring state estimation.

The utility of the method presented here depends, as do Kalman filter-based methods, upon the definition of data windows that permit the identification of feasible tracks and the elimination of infeasible tracks in a timely manner and according to statistically meaningful criteria. After identifying feasible tracks, the Kalman filter can be utilized to obtain the required state estimates and to analyze and refine the information in the feasible track files.

1.0 INTRODUCTION TO THE PROBLEM

The problem of tracking multiple targets using measurements whose source is questionable seems to be intrinsic to surveillance systems. Problems having this character have received considerable attention since Sittler's paper [1] in 1964. A recent survey paper [2] by Bar-shalom has described the problem and has reviewed the literature and principal results regarding the solution of the problem. Our discussion will assume familiarity with this body of work to at least the depth provided by Reference 2.

The problem that is considered here has the following description. Measurements are obtained at discrete times; t_1, t_2, $\cdots$, t_k, $\cdots$. At each measurement time t_k, a collection of measurements

$$\underline{z}_k = \left\{ \underline{z}_i(t_k);\ i = 1, 2, \cdots, M_k \right\},\ k = 1, 2, \cdots$$

is obtained. Each measurement $\underline{z}_i(t_k)$ is generated by a different source. For this discussion, the set of measurements up to and including $\underline{z}_k$ is denoted as

$$\underline{z}^k = \left\{ \underline{z}_i;\ i = 1, 2, \cdots, k \right\}$$

No ordering of measurements is preserved, necessarily, from one sampling time to another. For example, $\underline{z}_1(t_1)$ and $\underline{z}_1(t_2)$ may emanate from entirely different sources with the result that they are completely independent of each other. It is this lack of ordering that produces that multitarget tracking problem that is to be addressed here.

If there were no uncertainty regarding the source of each measurement, the multitarget tracking problem would be solved by a straightforward application of the methods developed for single-target tracking (e.g., extended Kalman filter). As described in

Reference 2, previous approaches to the problem have, generally, been based on state-space models and recursive filtering methods. The new aspects of the multitarget tracking problem stem from the necessity for measurement-to-track assignment of data and for the incorporation into the filtering algorithm of the possibility of incorrect data assignment. It is to this track assignment problem that attention is addressed in this report.

To provide a basis for the discussion consider the following mathematical model for the problem. Suppose that M_k, $\{k = 1, 2, \cdots\}$ measurement vectors $\underline{z}_i(t_k)$, $\{i = 1, 2, \cdots, M_k\}$ are obtained at measurement time t_k. These measurements are assumed to emanate from <u>different</u> sources. Thus, there are M_k sources at t_k. However, only a subset of these sources have interest to the surveillance system although the interest in a source is to be deduced from the analysis of the data. Briefly, each measurement $\underline{z}_i(t_k)$ may be either a signal $\underline{y}_i(t_k)$ plus noise $\underline{v}_i(t_k)$ or noise $\underline{v}_i(t_k)$, alone. The solution of this detection problem is fundamental to the satisfactory performance of the surveillance system.

The detection is accomplished through the analysis of data $\underline{z}^k$ obtained over some interval $[t_1, t_k]$. Generally, sources of interest (i.e., targets) exhibit motion during the observation interval. The motion is characterized by defining a state-space model that is based on a differential or a difference equation. For this analysis, the j^{th} target is assumed to be described by a state vector $\underline{x}$, satisfying a linear difference equation

$$\underline{x}_j(t_{k+1}) = \Phi \underline{x}_j(t_k) + \Gamma \underline{u}_j(t_k) + \underline{w}_j(t_k) \quad . \tag{1.1}$$

The vector $\underline{u}_j$ represents a control input to the system which is unknown to the surveillance system but cannot be neglected (e.g., heading changes). The vector $\underline{w}_j$ represents random influences on the system motion and is assumed to be described as a white noise sequence with zero mean and known covariance matrix Q. The matrices Φ, Γ, and Q are assumed to be known and time-invariant. The time-invariance of the system is a restriction on the generality of the problem described by Bar-shalom [2] in his survey paper. Nonetheless, it does describe an important class of problems that, based on [2], has not been treated extensively.

The i^{th} measurement at t_k is related to the target state in the following manner

$$\underline{z}_i(t_k) = H\,\underline{x}_j(t_k) + \underline{v}_i(t_k) \tag{1.2}$$

The matrix H is known and time-invariant. The noise $\underline{v}_i$ is assumed to be zero mean, white noise with time-invariant covariance matrix R. Measurements $\underline{z}_i(t_k)$ that are not generated by a target shall be assumed here to be independent and identically distributed random variables drawn from a uniform distribution defined on the measurement space.

In Equation (1.2), the i^{th} measurement $\underline{z}_i(t_k)$ is indicated as being related to the state of the j^{th} target. If the specific target were known, the multitarget tracking problem would be readily solved. The problem to be addressed here relates to the data association problem of establishing the specific target j to which the i^{th} measurement relates.

Several approaches have been taken to the solution of the data association problem. Basic to most of these is the application of Kalman filter for state estimation. The Kalman filter is used to estimate the state at t_k from the data assigned to a candidate track. Then, this estimate is used to predict the next measurement $\underline{z}(t_{k+1})$. Typically, some type of data window is defined to establish the next measurement that is to be assigned to the track. When more than one measurement could be assigned, various alternatives arise ranging from accepting only a single "or best" measurement to "splitting" the track into several candidates. Since the possibility exists that a measurement might be mistakenly assigned to a track, some researchers have suggested modifications to the Kalman filter to reflect the possible misassignment. It is important to recognize that several different Kalman filters may operate at each time; several more than the number of actual target tracks. Thus, computational burden can become an important concern. Several researchers have explored methods for establishing "feasible" tracks or "most likely" tracks as a screening mechanism to eliminate tracks to reduce the computational burden.

In the next section, an approach to the problem of determining feasible tracks that seem to be extremely efficient for the class of linear, time-invariant systems defined above. In brief, the approach is based on the recognition that input-output models can be defined that eliminates the need for state estimation to accomplish the data association problem. If state estimation can be deferred until after establishing feasible tracks, considerable unnecessary filter computation can be eliminated.

2.0 DATA ASSOCIATION USING INPUT-OUTPUT MODELS

Consider the dynamic and measurement model for a specific target. For convenience, the subscripts i and j appearing in Equations (1.1) and (1.2) are omitted.

$$\underline{x}(k+1) = \Phi \underline{x}(k) + \Gamma \underline{u}(k) + \underline{w}(k) \tag{2.1}$$

$$\underline{z}(k) = H\underline{x}(k) + \underline{v}(k) \tag{2.2}$$

It is straightforward to eliminate the state variables and to determine a difference equation model involving only the inputs and outputs when the target is operating in steady-state. By eliminating the state variables, the need for estimating them is eliminated. The input-output model can be used directly to predict measurements at future times. The variance of the error in the predictions can be determined and used to establish feasible tracks.

To establish an input-output model, let q denote a forward difference operator.

$$\underline{x}(k+1) \triangleq q\underline{x}(k) \tag{2.3}$$

Then, (2.1) can be written as

$$(qI-\Phi)\underline{x}(k) = \Gamma \underline{u}(k) + \underline{w}(k)$$

Solving for $\underline{x}(k)$, one obtains

$$\underline{x}(k) = (qI-\Phi)^{-1}[\Gamma \underline{u}(k) + \underline{w}(k)]$$

Using this result in (2.2), it follows* that

---

*The definition $A^{-1} = \frac{\text{adj}\, A}{\det A}$ is used.

ORINCON

$$\underline{z}(k) = H(qI-\Phi)^{-1}[\Gamma\underline{u}(k) + \underline{w}(k)]$$

or

$$\det(qI-\Phi)\underline{Z}(k) = H\,\mathrm{adj}(qI-\Phi)[\Gamma\underline{u}(k) + \underline{w}(k)]$$

$$+ \det(qI-\Phi)\underline{v}(k) \tag{2.4}$$

where adj$(qI-\Phi)$ denotes the adjoint of the matrix $(qI-\Phi)$ and det$(qI-\Phi)$ denotes the determinant of the matrix $(qI-\Phi)$.

In Equation (2.4), it is important to recognize that det$(qI-\Phi)$ represents a polynomial of n^{th} order in terms of the forward operator q. Thus, the left-hand side of (2.4) is a n^{th} order difference equation involving the output (or measurement) vectors $\underline{z}$. Furthermore, the elements of the adjoint matrix are polynomials of no more than $(n-1)^{st}$ order in q. Thus, the right-hand side of (2.4) is a difference equation involving the input vectors $\underline{u}$ and $\underline{w}$.

Example: To illustrate the preceeding discussion, suppose the dynamic model is based on planar motion at a nominally constant velocity except for random forcing function. Suppose, also, that noisy measurements are available of the position of the target. The state space model has the form

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \end{bmatrix} = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} + \begin{bmatrix} w_1(k) \\ w_2(k) \\ w_3(k) \\ w_4(k) \end{bmatrix} \tag{2.5a}$$

or

$$\underline{x}(k+1) = \Phi\underline{x}(k) + \underline{w}(k) \tag{2.5b}$$

The measurement model is

$$\begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} = \begin{bmatrix} x_1(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} v_1(k) \\ v_2(k) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} + \begin{bmatrix} v_1(k) \\ v_2(k) \end{bmatrix} \tag{2.6a}$$

or

$$\underline{z}(k) = H\underline{x}(k) + \underline{v}(k) \tag{2.6b}$$

From (2.5), the matrix $(qI-\Phi)$ is given by

$$qI-\Phi = \begin{pmatrix} q-1 & -\Delta t & 0 & 0 \\ 0 & q-1 & 0 & 0 \\ 0 & 0 & q-1 & -\Delta t \\ 0 & 0 & 0 & q-1 \end{pmatrix}$$

It follows that

$$\det(qI-\Phi) = (q-1)^4$$

$$\operatorname{adj}(qI-\Phi) = \begin{pmatrix} (q-1)^3 & (q-1)^2\Delta t & 0 & 0 \\ 0 & (q-1)^3 & 0 & 0 \\ 0 & 0 & (q-1)^3 & (q-1)^2\Delta t \\ 0 & 0 & 0 & (q-1)^3 \end{pmatrix}$$

and

$$(qI - \phi)^{-1} = \begin{pmatrix} \frac{1}{q-1} & \frac{\Delta t}{(q-1)^2} & 0 & 0 \\ 0 & \frac{1}{q-1} & 0 & 0 \\ 0 & 0 & \frac{1}{q-1} & \frac{\Delta t}{(q-1)^2} \\ 0 & 0 & 0 & \frac{1}{q-1} \end{pmatrix}$$

Using these results in (2.4), we have

$$(q-1)^4 \underline{z}(k) = \begin{pmatrix} (q-1)^3 & (q-1)^2\Delta t & 0 & 0 \\ & & & \end{pmatrix} \begin{pmatrix} w_1(k) \\ w_2(k) \\ w_3(k) \\ w_4(k) \end{pmatrix} + (q-1)^4 \begin{pmatrix} v_1(k) \\ v_2(k) \end{pmatrix}$$

Thus, a fourth-order difference equation for the outputs, a third-order difference equation for the plant inputs, and fourth-order difference equation for the measurement errors are obtained.

From the definition of the system, there are actually two, uncoupled, second-order systems being considered. This fact is reflected by noting that $(q-1)^2$ can be eliminated from the model to obtain

$$(q-1)^2 \underline{z}(k) = \underline{z}(k+2) - 2\underline{z}(k+1) + \underline{z}(k)$$

$$= \begin{pmatrix} w_1(k+1) - w_1(k) + \Delta t w_2(k) \\ w_3(k+1) - w_3(k) + \Delta t w_4(k) \end{pmatrix} + \underline{v}(k+2) - 2\underline{v}(k+1) + \underline{v}(k)$$

$$\underline{z}(k+2) - 2\underline{z}(k+1) + \underline{z}(k) = \underline{v}(k+2) + [B\underline{w}(k+1) - 2\underline{v}(k+1)]$$

$$+ [C\underline{w}(k) + \underline{v}(k)] \tag{2.7}$$

where $B \triangleq \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

$C \triangleq \begin{bmatrix} 0 & \Delta t & 0 & 0 \\ 0 & 0 & 0 & \Delta t \end{bmatrix}$

Equation (2.7) is a two-dimensional description of the behavior of the measurements as a function of the random plant inputs and the measurement noise. Again, it is important to note that there is no dependence upon the state with the result that no state estimation is required.

Using the input-output model (2.4), predictions of future measurements can be obtained. Since the covariance of the random noise is known, the variances of the prediction errors can be determined. There are two major problems that must be solved to utilize (2.4) for prediction.

(1) White noise estimation: From the data, estimates of the noise variables appearing on the RHS of (2.4) can be made. These estimated values are then used in predicting future output values. This problem is discussed in Section 3.

(2) Initiating the prediction: Special considerations are required to initiate the predictions and the white noise estimation. This problem is discussed in Section 4.

3.0 WHITE NOISE ESTIMATION

The predictive model (2.4) can be rewritten as

$$\underline{z}(k) + A_1\underline{z}(k-1) + \cdots + A_n\underline{z}(k-n)$$

$$= B_1\underline{u}(k-1) + \cdots + B_n\underline{u}(k-n)$$

$$+ C_1\underline{w}(k-1) + \cdots + C_n\underline{w}(k-n)$$

$$+ \underline{v}(k) + A_1\underline{v}(k-1) + \cdots + A_n\underline{v}(k-n) \tag{3.1}$$

The matrices A_i, B_i, C_i, $\{i = 1, 2, \cdots, n\}$ are known and their definitions follow by inspection of Equation (2.4). For the example, it is apparent from (2.7) that $n = 2$ and

$$A_1 = -2I, \quad A_2 = I, \quad C_1 = B, \quad C_2 = C, \quad B_1 = B_2 = 0$$

As discussed in Appendix A, the model (3.1) can be simplified by modifying the noise sequence on the RHS [3]. Further, the plant control inputs $\underline{u}(i)$ shall be neglected for the remainder of this discussion since they are regarded as unknown but nonrandom. The noise sequences $\underline{w}(i)$, $\underline{v}(i)$ can be replaced by a single noise sequence, say $\underline{n}(i)$, having the same statistical properties and with the property that the model is invertible. The procedure for determining the modified model is given in Appendix A. Thus, the matrices D_i, $(i = 1, 2, \cdots, n)$ and the covariance matrix N are assumed to be known and (3.1) becomes

$$\underline{z}(k) + A_1\underline{z}(k-1) + \cdots + A_n\underline{z}(k-n)$$

$$= \underline{n}(k) + D_1\underline{n}(k-1) + \cdots + D_n\underline{n}(k-n) \tag{3.2}$$

where $\underline{n}(i)$ represents a zero mean, white noise sequence with covariance matrix N. The model (3.2) provides the basis for the remainder of our discussion.

Consider the problem of predicting $\underline{z}(k)$, given measurements $\underline{z}(k-1)$, $\underline{z}(k-2)$, $\cdots$. It is well known [4] that the estimator that minimizes the mean-square error is the conditional expectation

$$\hat{\underline{z}}(k/k-1) \triangleq E[\underline{z}(k)/\underline{z}(k-1), \underline{z}(k-2), \cdots] \tag{3.3}$$

Using (3.2), the predictor is given by

$$\begin{aligned}\hat{\underline{z}}(k/k-1) = & -A_1\underline{z}(k-1) - \cdots - A_n\underline{z}(k-n) + \hat{\underline{n}}(k/k-1) \\ & + D_1\hat{\underline{n}}(k-1/k-1) + \cdots + D_n\hat{\underline{n}}(k-n/k-1)\end{aligned} \tag{3.4}$$

As in (3.3), the notational convention

$$\hat{\underline{n}}(j/\ell) \triangleq E[\underline{n}(j)|\underline{z}(\ell), \underline{z}(\ell-1), \cdots]$$

is used in (3.4). From (3.4), it is apparent that estimates of the white noise variables are required to evaluate the predictor. The white noise estimation problem is considered in the following paragraphs.

The noise $\underline{n}(i)$ is zero mean and white. Thus, one sees that

$$\begin{aligned}\hat{\underline{n}}(k/k-\ell) &= E[\underline{n}(k)|\, \underline{z}(k-\ell), \underline{z}(k-\ell-1), \cdots] \\ &= E[\underline{n}(k)] \\ &= 0\,, \quad \ell \geq 1\end{aligned}$$

It remains to consider $\hat{\underline{n}}(k-i/k-1)$, $i \geq 1$. Given a semi-infinite span of data, the white noise variables can be determined precisely since the noise model is invertible. To verify this statement, one solves the difference equation in (3.2) for $\underline{n}(k)$ as a function of the semi-infinite span of measurement values $\underline{z}(k)$, $\underline{z}(k-1)$, $\cdots$. Formally, this can be done by defining the operator

$$D(q^{-1}) = I + q^{-1}D_1 + q^{-2}D_2 + \cdots + q^{-n}D_n$$

$$A(q^{-1}) = I + q^{-1}A_1 = q^{-2}A_2 + \cdots + q^{-n}A_n$$

in terms of the operator q^{-1}

$$q^{-1}n(k) \triangleq n(k-1).$$

Then, Equation (3.2) becomes

$$A(q^{-1})\underline{z}(k) = D(q^{-1})\underline{n}(k) \tag{3.5}$$

Since $D(q^{-1})$ is invertible, the solution of (3.5) for $\underline{n}(k)$ is

$$\underline{n}(k) = \left[D(q^{-1})\right]^{-1} A(q^{-1})\underline{z}(k)$$

$$= \frac{1}{\det D(q^{-1})} \operatorname{adj} D(q^{-1})A(q^{-1})\underline{z}(k)$$

But $\det D(q^{-1})$ is a polynomial of n^{th} order in the delay operator q^{-1}. This implies that the inverse, $1/\det D(q^{-1})$, is an infinite series in q^{-1}. The invertibility of $D(q^{-1})$ ensures the convergence of the infinite series. Thus, $\underline{n}(k)$ has been expressed in terms of the semi-infinite collection of measurements. Obviously,

$$\hat{\underline{n}}(k/k) = E[\underline{n}(k)/\underline{z}(k), \underline{z}(k-1), \cdots]$$

$$= \underline{n}(k)$$

It follows, immediately, that

$$\underline{\hat{n}}(k-\ell/k) = \underline{n}(k-\ell), \quad \ell \geq 0$$

Assuming the availability of the data $\underline{z}(i)$, $(i = k-1, k-2, \cdots)$, the predictor (3.4) can be expressed as

$$\underline{\hat{z}}(k/k-1) = A_1\underline{z}(k-1) - \cdots - A_n\underline{z}(k-n)$$

$$+ D_1\underline{n}(k-1) + \cdots + D_n\underline{n}(k-n) \tag{3.6}$$

Consider the error in this predictor

$$\underline{\tilde{z}}(k/k-1) \triangleq \underline{z}(k) - \underline{\hat{z}}(k/k-1)$$

Comparing (3.6) and (3.2), it follows that

$$\underline{\tilde{z}}(k/k-1) = \underline{n}(k) \tag{3.7}$$

Note that (3.7) provides a convenient means for computing $\underline{n}(k)$ upon receipt of the measurement $\underline{z}(k)$. The covariance of the error in the estimate obtains immediately.

$$E[\underline{\tilde{z}}(k/k-1)\underline{\tilde{z}}^T(k/k-1)] = N$$

Unfortunately, a semi-infinite number of measurements is never available and the noise sequence $\underline{n}(k)$ must be estimated by other means. A reasonable estimator $\underline{\bar{n}}(k/k)$ can be postulated directly from (3.2)

$$\underline{\bar{n}}(k/k) \triangleq \underline{z}(k) + \sum_{i=1}^{n} A_i\underline{z}(k-i) - \sum_{i=1}^{n} D_i\underline{\bar{n}}(k-i/k-i) \tag{3.8}$$

This estimator requires knowledge of $\underline{\bar{n}}(k-i/k-i)$, $(i = 1, 2, \cdots, n)$.

The estimator $\underline{\bar{n}}(k/k)$ can be written as

$$D(q^{-1})\underline{\bar{n}}(k/k) = A(q^{-1})\underline{z}(k) \tag{3.9}$$

Subtracting (3.9) from (3.5) yields

$$D(q^{-1})[\underline{n}(k) - \underline{\bar{n}}(k/k)] = 0 \tag{3.10}$$

Since $D(q^{-1})$ is invertible (i.e., stable), it follows that the estimation error $\underline{n}(k) - \underline{\bar{n}}(k/k)$, must vanish as k becomes large without bound. Thus, $\underline{\bar{n}}(k/k)$, as given by (3.8), provides a reasonable estimator of $\underline{n}(k)$.

The covariance of the error in the estimate $\underline{\bar{n}}(k/k)$ is easily calculated using (3.10). Initial conditions for the covariance of the errors $\underline{\tilde{n}}(i/i)$, $i = k-1, \cdots, k-n$, are required. For this development, it shall be assumed that the errors in the estimates are uncorrelated. Since $\underline{n}(k)$ is white noise and since $\underline{\bar{n}}(k/k)$ converges to $\underline{n}(k)$, the errors must tend to be uncorrelated. From (3.10), the error $\underline{\tilde{n}}(k/k)$ can be written as

$$\underline{\tilde{n}}(k/k) = -D_1\underline{\tilde{n}}(k-1/k-1) - \cdots - D_n\underline{\tilde{n}}(k-n/k-n)$$

If the initial conditions are chosen to be unbiased

$$E[\underline{\tilde{n}}(k-i/k-i)] = 0, \; i = 1, 2, \cdots, n$$

then $\underline{\tilde{n}}(k/k)$ will be unbiased for all k. Thus, the covariance of the error is

$$\begin{aligned} E[\underline{\tilde{n}}(k/k)\underline{\tilde{n}}^T(k/k)] &= \sum_{i=1}^{n} D_i E[\underline{\tilde{n}}(k-i/k-i)\underline{\tilde{n}}^T(k-i/k-i)] \\ &\triangleq P_n(k/k) \\ &= \sum_{i=1}^{n} D_i P_n(k-i/k-i) D_i^T \end{aligned} \tag{3.11}$$

Remember that $P_n(k/k) \longrightarrow 0$ as $k \longrightarrow 0$.

Consider, now, the error in the predictor using $\bar{\underline{n}}$ in place of $\underline{n}$ in (3.6)

$$
\begin{aligned}
\underline{z}(k) - \hat{\underline{z}}(k/k-1) &= -\sum_{i=1}^{n} A_i \underline{z}(k-i) + \sum_{i=0}^{n} D_i \underline{n}(k-i) \\
&\quad + \sum_{i=1}^{n} A_i \underline{z}(k-i) - \sum_{i=1}^{n} D_i \bar{\underline{n}}(k-i/k-i) \\
&= \underline{n}(k) + \sum_{i=1}^{n} D_i [\underline{n}(k-i) - \bar{\underline{n}}(k-i/k-i)] \qquad (3.12)
\end{aligned}
$$

From the assumptions, it is clear that $\underline{z}(k/k-1)$ is an unbiased estimator. The covariance of the error is given by

$$
E \left\{ [\underline{z}(k) - \hat{\underline{z}}(k/k-1)] \, [\underline{z}(k) - \hat{\underline{z}}(k/k-1)]^T \right\} = N + P_n(k/k) \qquad (3.13)
$$

The definition of the predictor is now complete except for the consideration of initial conditions for the predictor. This aspect is considered in Section 4.

4.0 TRACK INITIATION

Suppose that the process is initiated with the measurement $\underline{z}_1$. But the predictor

$$\hat{\underline{z}}(k/k-1) = -\sum_{i=1}^{n} A_i \underline{z}(k-i) + \sum_{i=1}^{n} D_i \underline{\bar{n}}(k-i/k-i) \tag{4.1}$$

requires n measurements for the one-stage prediction. It also requires n white noise estimates.

In the absence of the requisite data, the available information reduces to the a priori statistical description of the random variables. By definition,

$$E[\underline{z}(k)] = 0 \quad \text{for all } k$$

$$E[\underline{n}(k)] = 0 \quad \text{for all } k$$

The covariance of the noise is known to be N. The covariance of the measurement $\underline{z}(k)$ is determined from the state space model (2.1)-(2.2)

$$\underline{Z}(k) \triangleq E[\underline{z}(k)\underline{z}^T(k)] = HE[\underline{x}(k)\underline{x}^T(k)]H^T + R \tag{4.2}$$

where

$$E[\underline{x}(k)\underline{x}^T(k)] = \Phi E[\underline{x}(k-1)\underline{x}^T(k-1)]\Phi^T + Q$$

let

$$M_k \triangleq E[\underline{x}(k)\underline{x}^T(k)]$$

Note that it satisfies the difference equation

$$M_k = \Phi M_{k-1} \Phi^T + Q$$

where an initial condition, say M_1, must be specified

$$E[\underline{x}(1)\underline{x}^T(1)] = M_1$$

Consider the prediction of $\underline{z}(2)$ from $\underline{z}(1)$. Then, (4.1) reduces to

$$\hat{\underline{z}}(2|1) = -A_1\underline{z}(1) + D_1\bar{\underline{n}}(1|1) \tag{4.3}$$

This represents an unbiased estimate where the a priori expected values of $\underline{z}(k-i)$, $\bar{\underline{n}}(k-i/k-1)$, $(i = 2, 3, \cdots, 4)$ have been used. The error in this predictor is

$$\begin{aligned}\tilde{\underline{z}}(2|1) &= -\sum_{i=1}^{n} A_i\underline{z}(2-i) + \sum_{i=1}^{n} D_i\underline{n}(2-i) + A_1\underline{z}(1) - D_1\bar{\underline{n}}(1|1)\\ &= -\sum_{i=2}^{n} A_i\underline{z}(2-i) + \sum_{i=2}^{n} D_i\underline{n}(2-i) + D_1[\underline{n}(1) - \bar{\underline{n}}(1|1)]\end{aligned}$$

The covariance is computed to be

$$\begin{aligned}E[\tilde{\underline{z}}(2|1)\tilde{\underline{z}}^T(2|1)] &= \sum_{i=2}^{n} A_i Z(2-i) A_i^T\\ &+ \sum_{i=2}^{n} D_i N D_i^T + D_1 P_n(1|1) D_1^T\end{aligned} \tag{4.4}$$

The noise $\underline{n}(1)$ must be estimated. Referring to (3.8), the estimate is

$$\bar{\underline{n}}(1|1) = \underline{z}(1) \tag{4.5}$$

with covariance matrix

$$P_n(1|1) = E[\tilde{\underline{n}}(1|1)\tilde{\underline{n}}^T(1|1)] = \sum_{i=1}^{n} A_i Z(1-i) A_i^T + \sum_{i=1}^{n} D_i N D_i^T \tag{4.6}$$

This calculation completes the first computational cycle and the second measurement can be processed when obtained.

Suppose $\underline{z}(2)$ has been obtained and consider the prediction of $\underline{z}(3)$. The noise sample $\underline{n}(2)$ is estimated, using (3.7) as

$$\underline{\bar{n}}(2|2) = \underline{z}(2) - \underline{\hat{z}}(2|1) \tag{4.7}$$

with covariance matrix

$$P_n(2|2) = \sum_{i=2}^{n} A_i Z(2-i) A_i^T + D_1 P_n(1|1) D_1^T + \sum_{i=2}^{n} D_i N D_i^T \tag{4.8}$$

The predictor is given by

$$\underline{\hat{z}}(3|2) = -A_1\underline{z}(2) - A_2\underline{z}(1) + D_1\underline{\bar{n}}(2|2) + D_2\underline{\bar{n}}(1|1) \tag{4.9}$$

The predictor has the covariance matrix

$$E[\underline{\tilde{z}}(3|2)\underline{\tilde{z}}^T(3|2)] = \sum_{i=3}^{n} A_i Z(3-i) A_i^T + \sum_{i=1}^{2} D_i P_n(3-i/3-i) D_i^T + \sum_{i=3}^{n} D_i N D_i^T \tag{4.10}$$

Equations (4.7) - (4.10) define the second computational cycle.

The algorithm described for $k = 1$ and $k = 2$ generalizes readily for $k \leq n$. The general procedure is stated below. <u>For $k = 1, 2, \cdots, n$</u>

$$\underline{\bar{n}}(k/k) = \underline{z}(k) - \underline{\hat{z}}(k/k-1) \tag{4.11}$$

$$P_n(k/k) = \sum_{i=k}^{n} A_i Z(k-i) A_i^T + \sum_{i=k}^{n} D_i N D_i^T + \sum_{i=1}^{k-1} D_i P_n(k-i/k-i) D_i^T \tag{4.12}$$

$$\underline{\hat{z}}(k/k-1) = -\sum_{i=1}^{k-1} A_i\underline{z}(k-i) + \sum_{i=1}^{k-1} D_i\underline{\bar{n}}(k-i/k-i) \tag{4.13}$$

$$E[\tilde{\underline{z}}(k/k-1)\tilde{\underline{z}}^T(k/k-1)] = \sum_{i=k}^{n} A_i Z(k-i) A_i^T$$

$$+\sum_{i=1}^{k-1} D_i P_n(k-i/k-i) D_i^T$$

$$+\sum_{i=k}^{n} D_i N D_i^T \qquad (4.14)$$

where $\hat{\underline{z}}(1|0) \triangleq 0$.

The system (4.11) - (4.14) applies for $k > n$ by following the convention that summations $\sum_{i=k}^{n} (\cdot) = 0$ for $k > n$ and $\sum_{i=1}^{k-1} (\cdot) \equiv \sum_{i=1}^{n} (\cdot)$. Thus, the algorithm is completely defined by (4.11) - (4.14) for all k. The covariance matrix Z(k) is given by (4.2).

5.0 MULTITARGET TRACKING ALGORITHM

The prediction algorithm (4.11) - (4.14) provides the basis for data association for the multitarget tracking algorithm. In the following discussion, the data association algorithm is presented. The predictions utilized in the algorithm are assumed to be obtained from the algorithm (4.11) - (4.14). The simplicity of this algorithm provides a significant computational advantage relative to existing algorithms which utilize the Kalman filter algorithm to accomplish the predictions. No filter calculations are required here for track prediction. The track predictions are carried out in terms of output variables which are generally of lesser dimension than the state dimension. Having established feasible tracks using the input-output model, the Kalman filter can be used for state estimation for only the feasible tracks. Only the data association problem is addressed in the following paragraphs.

Consider a measurement time t_k and suppose measurements $\{\underline{z}_1(t_k), \underline{z}_2(t_k), \cdots, \underline{z}_{m_k}(t_k)\}$ are obtained. At t_{k-1}, suppose that there are ℓ_{k-1} <u>track files</u>. A track file is regarded as a collection of data and white noise estimates that are used to predict the next measurement sample for the track. These track files shall be denoted as $f_{k-1}{}^i(\underline{z}^{k-1}, \underline{n}^{k-1})$. $(i = 1, 2, \cdots, \ell_{k-1})$. Each track file can be used to generate a measurement prediction $\hat{\underline{z}}_i(k/k-1)$ with associated covariance

$$P_z{}^i(k/k-1) \triangleq E[\tilde{\underline{z}}_i(k/k-1)\tilde{\underline{z}}_i{}^T(k/k-1)]$$

The corresponding error ellipsoid can be used to define a data window. Measurements $\underline{z}_i(t_k)$ contained within the window are associated with the track file. For each track prediction, one of three occurrences is possible.

(1) No measurements $\underline{z}_i(t_k)$ are contained in the error ellipsoid of the j^{th} track file.

In this case, one can continue the track file by predicting to the next measurement time and repeating the test. Although only one-stage prediction has been discussed, the algorithm (4.11) - (4.14) is adapted in a simple manner to accomplish N-stage predictions.

Alternatively, the absence of feasible measurements may indicate the infeasibility of the track file with the result that the file is dropped from memory. The logic for establishing conditions for eliminating specific track files is dependent upon statistical thresholds that are defined by the system designer. The objective of data association is to reduce the number of potential target tracks. Thus, it is imperative that track files are eliminated whenever reasonable.

(2) One measurement $\underline{z}_i(t_k)$ is contained in the error ellipsoid of the j^{th} track file.

This circumstance provides clear indication of the need for continuing the track file. The measurement is added to the file and the track file is updated. Note that, generally, the data $\underline{z}_i(t_{k-n-1})$ is not needed for prediction. It may be necessary to retain this measurement sample for input to the state estimator in the event that this track file is identified as being feasible.

It can happen that the measurement $\underline{z}_i(t_k)$ lies in the data window for more than one track file. From physical considerations, a measurement must relate to a single target. However, additional data may be required before a unique assignment of $\underline{z}_i(t_k)$ can be achieved. Alternatively, the measurement may be arbitrarily assigned at t_k to a specific track based on other considerations. For example, it may be assigned to the track yielding the smallest prediction error $[\underline{z}_i(t_k) - \hat{\underline{z}}_j(t_k/t_{k-1})]$. Alternatively, it may be assigned to the track having the smallest error ellipsoid volume.

(3) More than one measurement is contained in the data window for the j^{th} track file.

At least two alternatives present themselves in this situation. First, the measurement yielding the smallest prediction error (e.g., smallest in the sense that the squared-residual $[\underline{z}_i(t_k) - \hat{\underline{z}}_j(t_k/t_{k-1})]^T [\underline{z}_i(t_k) - \hat{\underline{z}}_j(t_k/t_{k-1})]$ is least for all $\underline{z}_i(t_k)$ contained in the data window) may be added to the track file and all other measurements rejected. Alternatively, a splitting of track files may be introduced. The track file $f_{k-1}{}^j(\cdot,\cdot)$ may be used to generate ℓ_k additional track files which differ only through the ℓ_k measurements $\underline{z}_i(t_k)$ which are contained in the data window.

Some measurements $\underline{z}_i(t_k)$ might not be contained in the data window for any track file or might not be included in a track file when track splitting is not permitted. It is important that every measurement is considered as emanating from a potential track. Measurements not added to a track file should be used to initiate a new track file. As discussed in Section 4, the prediction for tracks containing fewer measurements than the order of the model uses a priori statistical information about the measurement and input noise covariances. Care must be taken in defining these covariances in order to prevent the data windows for subsequent predictions from being unreasonably large.

From the preceding discussion, one notes that at t_k there will be at least M_k track files. Generally, there will be many more track files than M_k. Potentially, there could be as many as $(\prod_{i=1}^{k} m_i)$ track files. As this could be an enormous number, it is imperative that infeasible tracks be eliminated as early as possible. This elimination occurs when no measurements are contained in a track data window for one or more sampling times.

A feasible track is identified when the number of measurements in a track file becomes sufficiently large (i.e., k > n). When a track is identified, the file is retained as long as relevant data is obtained. But, also, the data can be provided to a state estimator. In addition to obtaining estimates of the state of the track, the feasibility of the track can be confirmed or refined by analysis of the resulting innovations sequence. This possibility shall only be noted here but shall not be pursued further.

A general algorithm has been defined for accomplishing data association and feasible track identification for the multitarget tracking problem. For the class of linear, time-invariant systems that have been considered, it appears that this procedure may be considerably more computationally efficient than procedures reviewed in Reference 2 which are based upon the use of the Kalman filter. The utility of the procedure depends, as for the KF-based methods, upon the definition of data windows that permit the elimination of infeasible tracks in a statistically satisfactory manner. The need for state estimation to achieve data association has been eliminated and a simpler prediction using equivalent input-output models has been proposed.

Because of the inherent complexity of the multitarget tracking problem, further analysis of this algorithm must be accomplished using numerical experiments. The detailed computations required for this type of analysis could not be performed due to a lack of funds remaining in this contract. The approach that has been defined appears to be extremely promising. Additional contractual support to investigate this approach in greater depth seems warranted.

6.0 REFERENCES

1. Sittler, R. W., "An Optimal Data Association Problem in Surveillance Theory," IEEE Trans. M.I. Elect., MIL-8, 125-139, April 1964.

2. Bar-shalom, Y., "Tracking Methods in a Multitarget Environment," IEEE Trans. Auto Cont., AC-23, 618-626, August 1978.

3. Kashyap, R. L., Rao, A. R., Dynamic Stochastic Models from Empirical Data, Academic Press, New York, N.Y., 1976.

4. Sorenson, H. W., Parameter Estimation: Principles and Problems, Marcel Dekker, New York, N.Y., 1979.

APPENDIX A: SIMPLIFYING THE NOISE MODEL

The noise terms in the RHS of (3.1) involves two white noise sequences. To simplify the development of the predictor, it is desirable to introduce a separate noise sequence, say $\underline{n}(i)$, that yields the same noise characteristics as $\underline{w}(i)$ and $\underline{v}(i)$. That is, we want to determine coefficient matrices D_i such that

$$\underline{n}(k) + D_1\underline{n}(k-1) + \cdots + D_n\underline{n}(k-n) = \underline{v}(k) + A_1\underline{v}(k-1)$$

$$+ \cdots + A_n\underline{v}(k-n) + C_1\underline{w}(k-1) + \cdots + C_n\underline{w}(k-n)$$

or $$\underline{\zeta}(k) \overset{\Delta}{\equiv} \underline{\zeta}_1(k) + \underline{\zeta}_2(k) \tag{A.1}$$

$$E[\underline{\zeta}_1(k)\underline{\zeta}_1^T(k-j)] = E\left\{\left[\sum_{i=0}^{n} A_i\underline{v}(k-i)\right]\left[\sum_{i=0}^{n} A_i\underline{v}(k-i-j)\right]^T\right\}$$

$$= \sum_{i=j}^{n} A_i R A_{i-j}^T$$

$$E[\underline{\zeta}_2(k)\underline{\zeta}_2^T(k-j)] = E\left\{\left[\sum_{i=1}^{n} C_i\underline{w}(k-i)\right]\left[\sum_{i=1}^{n} C_i\underline{w}(k-i-j)\right]^T\right\}$$

$$= \sum_{i=j+1}^{n} C_i Q C_{i-j}^T$$

For the new sequence $\underline{\zeta}(k)$, it follows, since $\underline{w}$ and $\underline{v}$ are independent, that

$$E[\underline{\zeta}(k)\underline{\zeta}^T(k-j)] = E[\underline{\zeta}_1(k)\underline{\zeta}_1^T(k-n)] + E[\underline{\zeta}_2(k)\underline{\zeta}_2^T(k-n)]$$

$$= \sum_{i=j}^{n} A_i R A_{i-j}^T + \sum_{i=j+1}^{n} C_i Q C_{i-j}^T$$

$$= 0 \quad \text{for all } j > n \tag{A.2}$$

But we want

$$E[\underline{\zeta}(k)\underline{\zeta}^T(k-j)] = \sum_{i=j}^{n} D_i N D_{i-j}^T$$

$$= 0 \quad \text{for } j > n \qquad \text{(A.3)}$$

Equating coefficients in (A.2) and (A.3), one obtains (n + 1) matrix unknowns D_1, D_2, $\cdots$, D_n, N.

$$D_n N = A_n R$$

$$D_{n-1} N + D_n N D_1 = A_{n-1} R + C_n Q C_1^T \qquad \text{(A.4)}$$

$$\vdots$$

$$\sum_{i=0}^{n} D_i N D_i^T = \sum_{i=0}^{n} A_i R A_i^T + \sum_{i=1}^{n} C_i Q C_i^T$$

All of the terms on the RHS of (A.4) are known. It is possible to solve for the D_i, $i = 1, 2, \cdots, n$ and for N. More than one solution is possible, generally. Then, the solution is chosen to insure that the system is invertible in the sense that the zeros of the determinant of

$$D(q) = q^n I + q^{n-1} D_1 + \cdots + D_n$$

are all contained within the unit circle. With the satisfaction of this condition, the difference equation

$$D(q)\underline{n}(k) = 0$$

is stable and $\underline{n}(k) \longrightarrow 0$ as $k \longrightarrow \infty$.

EXAMPLE

As derived earlier, consider the noise model given by

$$\underline{\zeta}_1(k) + \underline{\zeta}_2(k) = \underline{v}(k) - 2\underline{v}(k-1) + \underline{v}(k-2) + B\underline{w}(k-1) + C\underline{w}(k-2)$$

where $n = 2$ and

$$R = \sigma_v^2 I$$

$$Q = \begin{pmatrix} \sigma_{11}^2 & 0 & 0 & 0 \\ 0 & \sigma_{22}^2 & 0 & 0 \\ 0 & 0 & \sigma_{11}^2 & 0 \\ 0 & 0 & 0 & \sigma_{22}^2 \end{pmatrix}$$

Consider (A.4). This system is given below.

$$D_2N = A_2R = R = \sigma_v^2 I$$

$$D_1N + D_2ND_1 = A_1R + C_2QC_1^T = -2\sigma^2 I$$

$$N + D_1ND_1^T + D_2ND_2^T = R + A_1RA_1^T + A_2RA_2^T + C_1QC_1^T + C_2QC_2^T$$

$$= \sigma_v^2 I + 4\sigma_v^2 I + \sigma_v^2 I + \sigma_{11}^2 I + \Delta t^2 \sigma_{22}^2 I$$

$$= \left[6\sigma_v^2 + \sigma_{11}^2 + \Delta t^2 \sigma_{22}^2\right] I$$

Substituting D_2N into the second and third equations, we obtain

$$D_1N + \sigma_v^2 D_1 = 2\sigma_v^2 I$$

$$N + D_1ND_1^T + \sigma_v^2 D_2^T = \left[6\sigma_v^2 + \sigma_{11}^2 + \Delta t^2 \sigma_{22}^2\right] I$$

$$D_1 \left(N + \sigma_v^2 I \right) = -2\sigma_v^2 I$$

$$D_1 = 2\sigma_v^2 \left[\sigma_v^2 I + N \right]^{-1}$$

$$D_2 = \sigma_v^2 N^{-1}$$

$$N + 4\sigma_v^4 \left[\sigma_v^2 I + N \right]^{-1} N \left[\sigma_v^2 I + N \right]^{-1} + \sigma_v^4 N^{-1} = \left(6\sigma_v^2 + \sigma_{11}{}^2 + \Delta t^2 \sigma_{22}{}^2 \right) I$$

$$N \left(\sigma_v^2 I + N \right) + 4\sigma_v^4 \left(\sigma_v^2 I + N \right)^{-1} N + \sigma_v^4 N^{-1} \left(\sigma_v^2 I + N \right) = a \left(\sigma_v^2 I + N \right)$$

$$N \left(\sigma_v^2 I + N \right)^2 + 4\sigma_v^4 N + \sigma_v^4 N^{-1} \left(\sigma_v^2 I + N \right)^2 = a \left(\sigma_v^2 I + N \right)^2$$

$$N^2 \left(\sigma_v^2 I + N \right)^2 + 4\sigma_v^4 N^2 + \sigma_v^4 \left(\sigma_v^2 I + N \right)^2 = a \left(\sigma_v^2 I + N \right)^2 N$$

$$N^2 \left(\sigma_v^4 I + 2\sigma_v^2 N + N^2 \right) + 4\sigma_v^2 N^2 + \sigma_v^4 \left(\sigma_v^4 I + 2\sigma_v^4 N + N^2 \right)$$

$$= a \left(\sigma_v^4 I + 2\sigma_v^2 N + N^2 \right) N$$

$$N^4 + 2\sigma_v^2 N^3 + \sigma_4^2 N^2 + 4\sigma_v^2 N^2 + \sigma_v^4 N^2 + 2\sigma_v^8 N + \sigma_v^8 I$$

$$= aN^3 + 2a\sigma_v^2 N^2 + a\sigma_v^4 N$$

or $$N^4 + \left(2\sigma_v^2 - a \right) N^3 + \left(\sigma_v^2 + 4\sigma_v^2 + \sigma_v^2 - 2a\sigma_v^2 \right) N^2$$

$$+ \left(2\sigma_v^8 - a\sigma_v^4 \right) N + \sigma_v^8 I = 0$$

This equation can be satisfied by assuming N to have the form

$$N = \eta I$$

Then, a scalar, quartic equation must be solved to complete the definition of the modified system.

$$\eta^4 + \left(2\sigma_v^2 - a\right)\eta^3 + \left(\sigma - 2a\right)\sigma_v^2\eta^2 + \left(2\sigma_v^4 - a\right)\sigma_v^4\eta + \sigma_v^8 = 0$$

The value of N is used to determine D_1 and D_2

$$D_1 = \frac{-2\sigma_v^2}{\sigma_v^2 + \eta} I$$

$$D_2 = \frac{\sigma_v^2}{\eta} I$$

SECTION III
A DISCUSSION OF THE OCEAN TACTICAL TARGETING PROBLEM

1.0 INTRODUCTION

Any realistic approach to the ocean targeting problem requires the interaction of many diverse sensor and computational systems; analytical and computational techniques; and many political empires. The fusion of these elements is fundamental to any solution to the OT^2 problem. This white paper is an attempt to describe the problem from a systems scientific point of view and to highlight problem areas.

Consider a large ocean surveillance area with well defined boundaries. This surveillance area contains surface and subsurface ships as well as aircraft and other signal sources. These may be threats, potential threats, friendlys or neutrals. Each of these categories could be broken down into finer divisions--such as combatant, merchant, fishing--ending in hull number or specific identification of all sources.

Resources including sensors, computational resources, communication resources and weapon resources are available to help monitor and control the ocean surveillance area. There are numerous sensors and sensor systems that gather data containing information about targets in the surveillance area or about passages into or out of the area. There are communication links that allow transmission of raw and processed sensor data; information about potential target tracks and identification; and information used to control or allocate sensors, computational, personnel and weapon resources.

There are computers used to perform the computational tasks of signal processing, data association, detection, tracking, classification and resource allocation. There are weapons to be allocated to threat targets and, last but not least, personnel to be allocated to various aspects of the problem requiring human interaction.

Thus the ocean targeting problem is the problem of utilizing these available resources to first obtain an accurate picture of the surveillance area and secondly to allow adequate weapon allocation to be made. This can be visualized as a classical control system with two feedback loops as shown in Figure 1. The feed-forward loop consists of the surveillance area and all that it contains: the sensors, sensor system processing and fusion processing required to do classical ocean surveillance.

The first or inner loop is the sensor control loop which allows control of the sensor resources. This allows the focusing of the sensor resources on a particular region of interest or a particular target of interest. Thus sensors can be focused on a particular geographical or frequency region or on events of interest because of *a priori* information, or information obtained from completely different sensor systems. The utilization of this inner loop in an intelligent manner is vital to the success of an ocean targeting system but it also implies great problems of a theoretical, computational and political nature. This is one of the major unsolved problems in development of a real ocean targeting system.

The second feedback in Figure 1 is the weapon allocation/fire control loop. This is the major role of command, control and communication systems. It should be clear that any design of a weapon allocation system will have great impact on the sensor allocation loop and the processing done in the feed-forward ocean surveillance path. In order to reduce the complexity of this discussion, we will first focus on the feed-forward path of the ocean tactical targeting problem without considering the feedback paths. The effects and problems introduced by the inner and outer feedback loops will be considered later.

The difficulties inherent in the feed-forward path or in the surveillance problem can be broken down into several areas. These include the standard multi-target, multi-sensor detection, tracking,

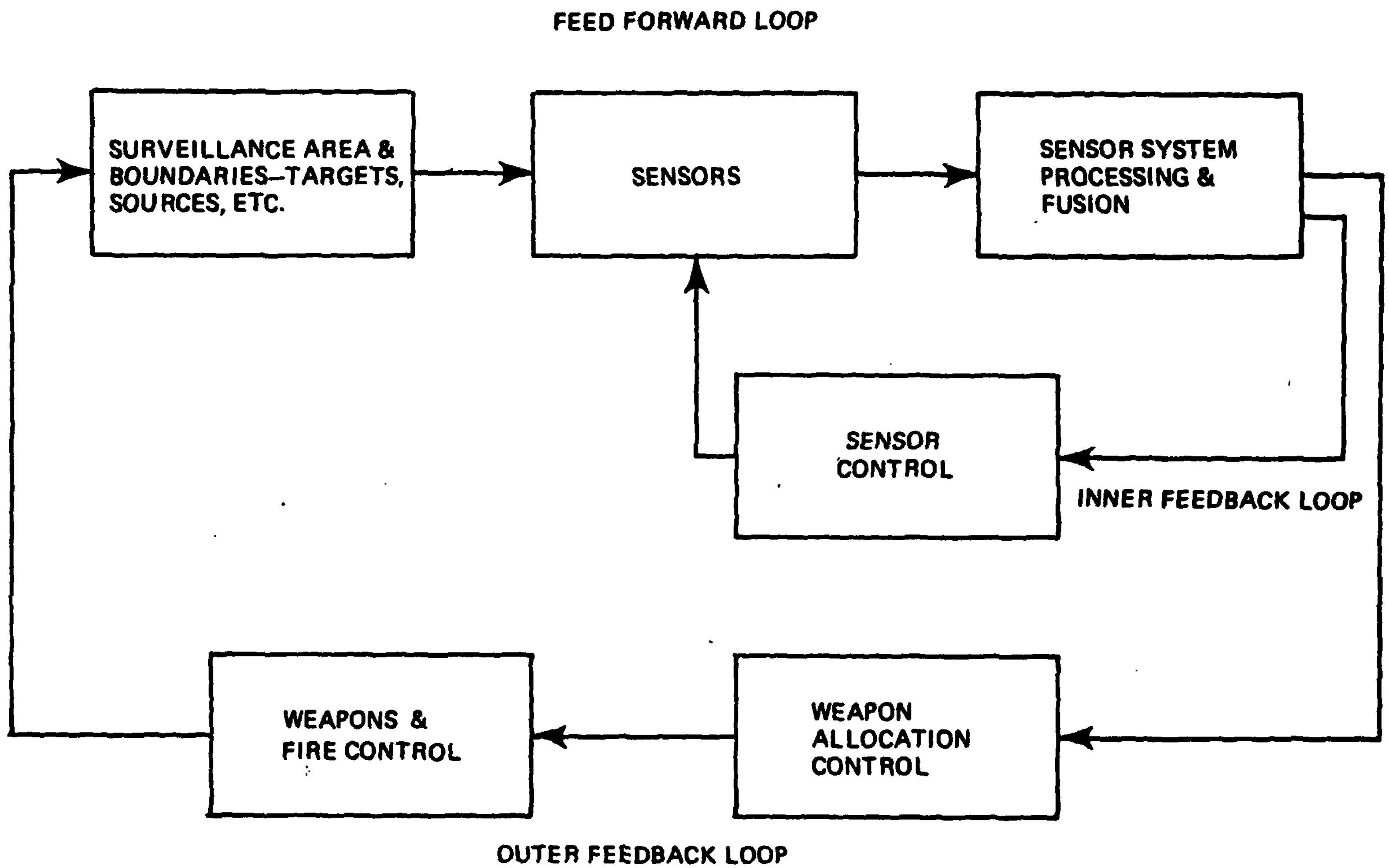


Figure 1. OT^2 as a feedback control system (inner and outer loop *observing* and *controlling ocean basin*).

and classification problem in clutter; the data communication problem; the distributed processing problem; the data base problem; the man/machine interface problem; and the political problem.

A somewhat more complex view of the OT^2 problem is shown in Figure 2. Here many possible levels of sensor system signal processing are shown. Also, both the inner and outer feedback possibilities from Figure 1 are indicated. Note that the inner sensor control feedback path is really a distributed feedback loop. Thus at least the possibility exists for sensor level control at each sensor, each sensor system level and at the overall fusion center. Sensor control could conceptually be exercised from the C^3 function in order to improve fire control or targeting functions.

There are many difficulties inherent in the feed-forward path or in the surveillance problem. Some of these, which ORINCON feels are key research areas, are discussed in the sections to follow.

Section 2 is a discussion of the overall ocean tactical targeting problem with particular consideration of various data partitions.

In Section 3 the problems and potential solutions to data communication will be addressed. Two approaches appear viable; namely, data compression and sensor level preprocessing.

Section 4 presents the key features pertaining to distributed information processing as it relates to ocean tactical targeting. It discusses a host of scenarios that could be implemented.

A number of problems in data base management will be illustrated in Section 5. It then goes on to describe certain techniques that could be applied to attack these problems.

In Section 6, a general discussion of the measurement fusion problem is presented. Following this, in Section 7, the multi-sensor/multi-target tracking, detection, and classification problem is treated.

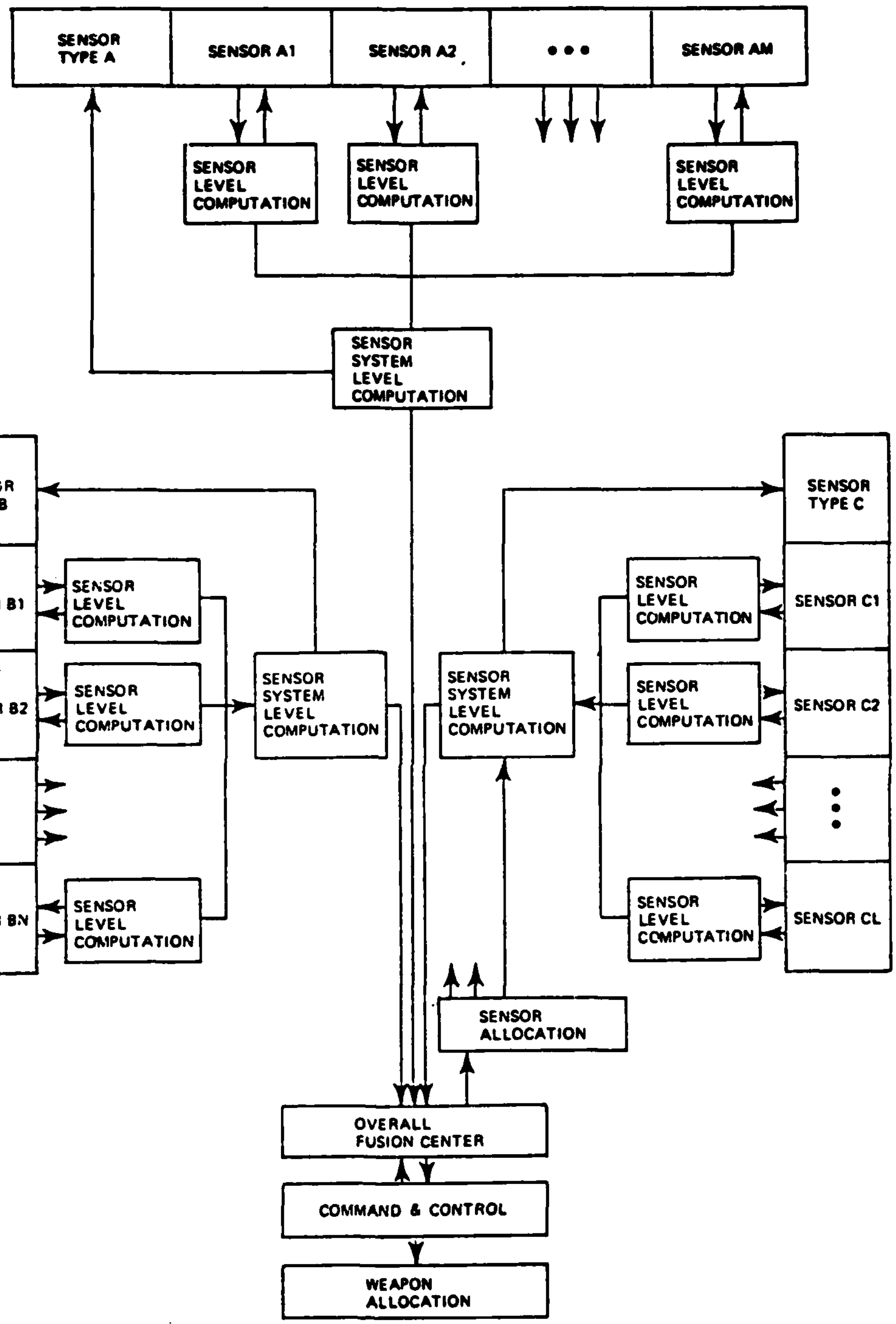


Figure 2. Ocean tactical targeting overview.

One of a number of possible approaches is discussed in some detail to illustrate the complexity of the measurement fusion problem.

Finally, in Sections 8 and 9, the issues of force and weapon allocation and final targeting will be addressed.

2.0 THE OCEAN SURVEILLANCE PROBLEM

The ocean surveillance problem as defined in the last section is to take and process data from all available sensors which contain information about the ocean surveillance area or passages through its boundary, and produce the best possible picture of the surveillance area. This requires the detection, tracking and classification of all possible targets and the allocation of the computational, communication and personnel resources available. This is summarized in an extremely simple manner in Figure 2, which shows the possible sensors arranged vertically. Any spatial factors such as beams, time delay and Doppler resolution cells, range and range rate cells or positional cells (of a particular sensor) are indicated by the horizontal "spatial sector" direction. The time flow of data for a given sensor and spatial sector (perhaps a beam) is indicated by the axes into the paper. Thus this figure simply indicates that the overall "open loop" surveillance problem consists of taking all the data from a suite of sensor systems over a time interval and drawing an accurate picture of the surveillance area. This accurate picture of the surveillance area must show the time evolution of the states of all potential targets. It requires that the functions of target detection, target tracking and target identification be done, as well as possible, from the sensor data available. The target state will include position and velocity information as well as any other information required to specify the future state of the target based on the current state and assumed dynamics. This could include frequency information, local sound speed information, and turns-per-knot information, as well as parameters of the system dynamics.

In actual practice it is impossible to take all of the information from the sensors and process it simultaneously to obtain an overall picture of the surveillance area or--as in Figure 2--an "all sensors master track file." However, it is clear that if this were possible, the maximum information could be extracted from the raw sensor information, and the best possible system performance could be obtained.

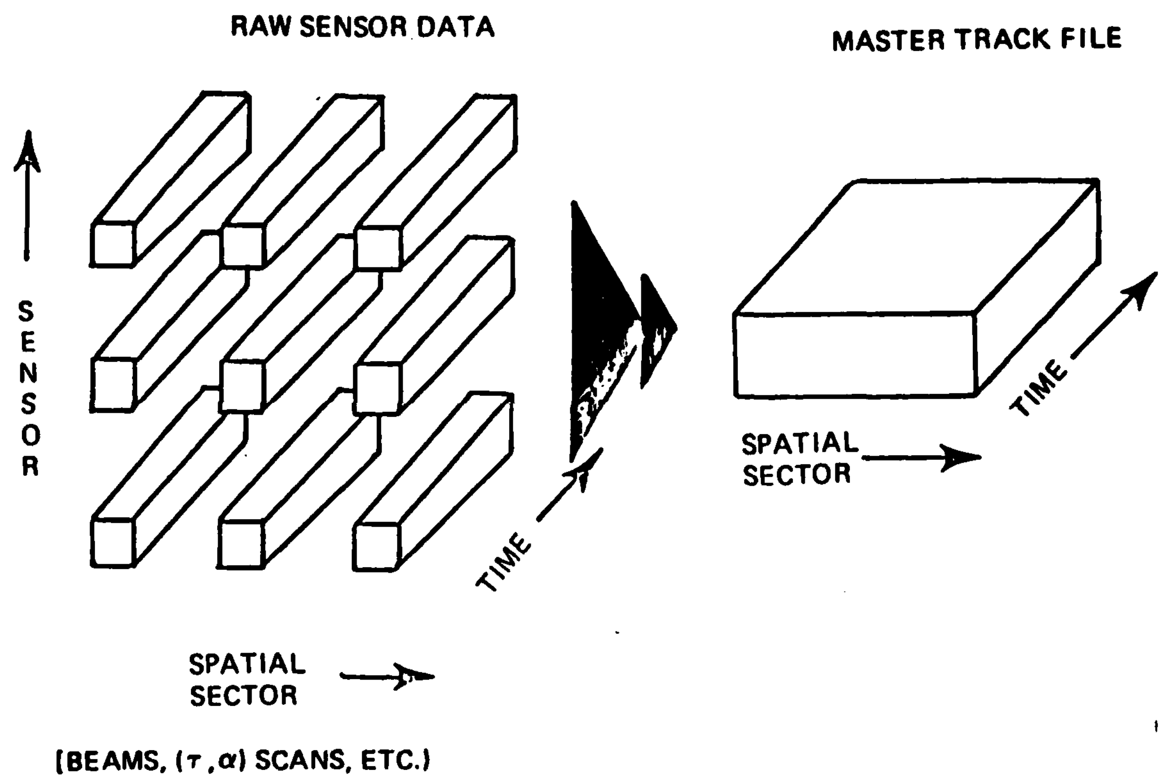


Figure 3. Raw data to master track file.

In any realistic attempt to solve this problem, intermediate processing steps must be performed. This is required to break the problem down into easily handled segments and is intended to reduce the overall amount of processing required, to reduce the amount of data transmitted, and to allow the staff working at a given sensor level to monitor the progress of the system. This will generally lead to performing operations at a local level when possible.

Thus, a realistic approach to the problem might be as indicated in Figure 3. Here data from each sensor system is processed repeatedly to form a master track file for that sensor system. This overall track information could then be merged in a fusion facility at the track level to form an all-sensor master track file. It should be noted that information could be lost in such a system. If two pieces of information from two different sensor systems were sufficient to form a good track when put together but not sufficient to give tracks in the separate sensor track files, then the track would be lost. Nonetheless, some loss of information will probably have to be tolerated in order to allow the global problem to be handled in even a sub-optimal manner.

In actual practice the current system looks more like Figure 4, where tracking is done on the output of each individual sensor (perhaps in sensor coordinates), and then all of the tracks in sensor coordinates were put together to form a sensor system master track file. The last step of combining the tracks from numerous sensor systems is done in a very crude manual fashion.

Note that the possibility for all of these levels of processing was contained in the more complex view of the OT^2 problem shown in Figure 2. There many of the possible levels of sensor system processing are shown as in Figure 4.

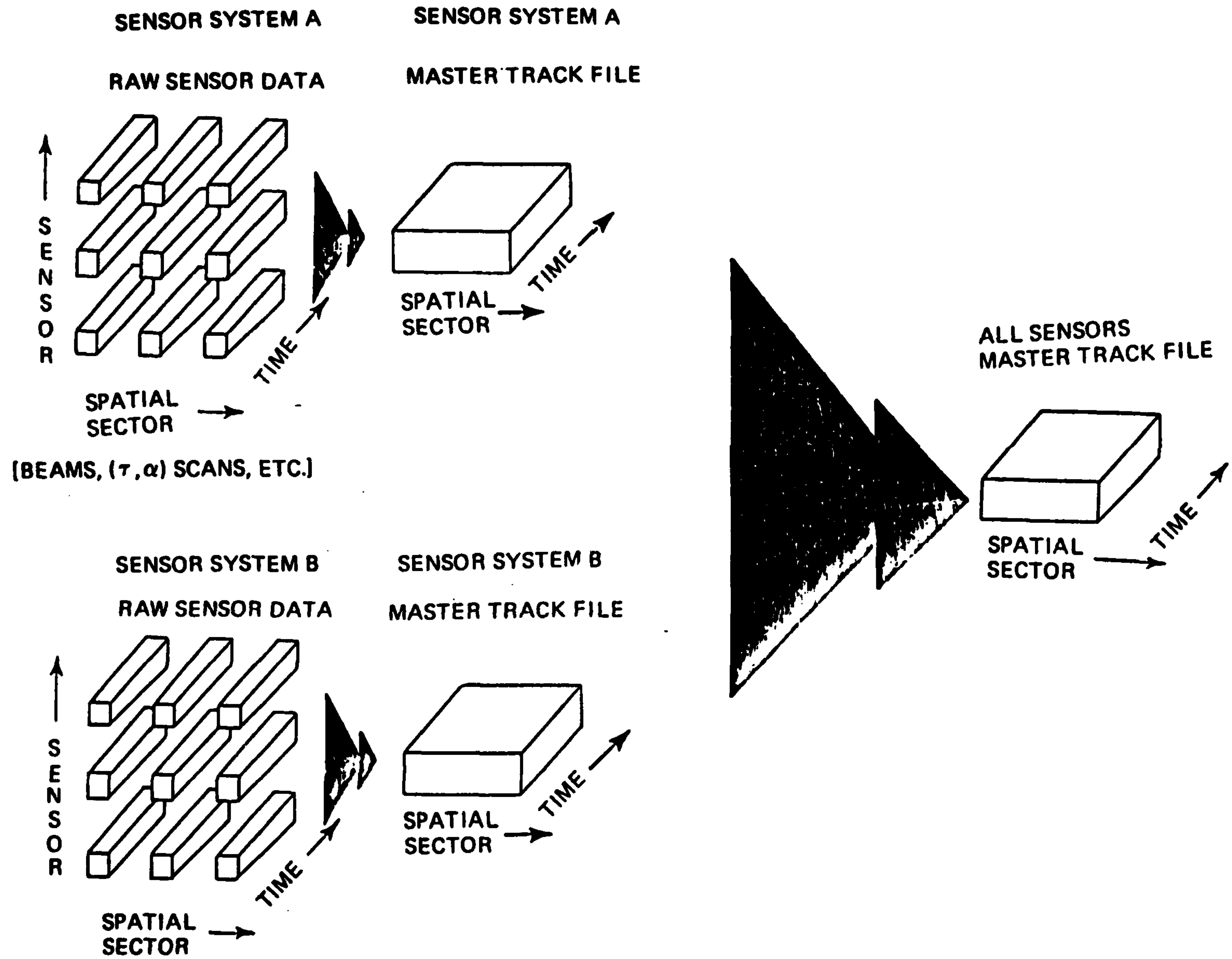


Figure 4. Sensor system tracks to master track file.

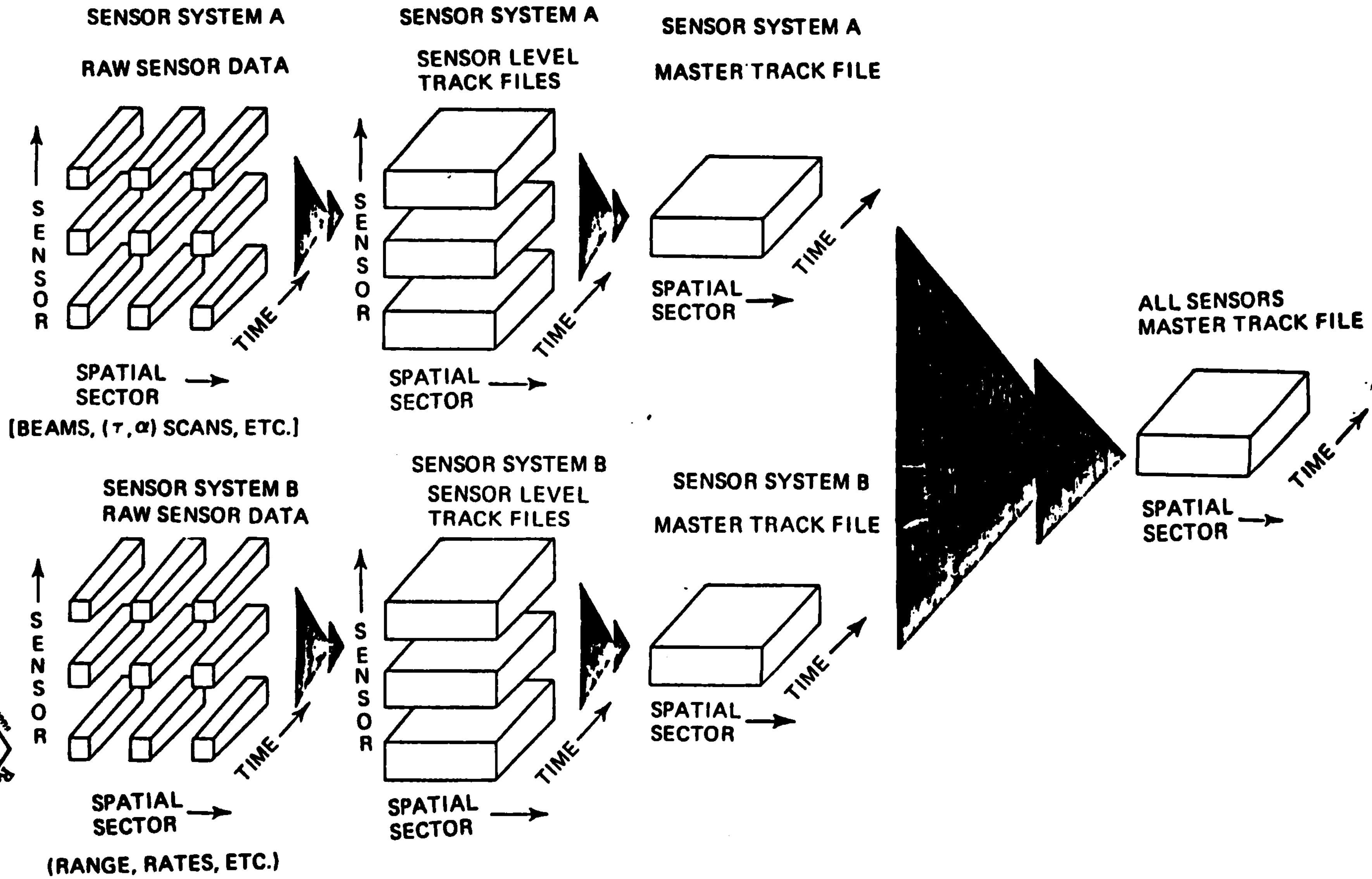


Figure 5. Sensor level to sensor system track files to master track file.

3.0 DATA COMMUNICATION PROBLEMS

The data communications problem is of paramount importance in a large, distributed surveillance network. The raw information rates produced by sensor systems in the network will easily overwhelm the relatively low capacity transmission links that connect the network nodes. Thus the network structure, and particularly the distribution of processing resources, will be fundamentally constrained by the data communication resources available. The two principal qualities of these resources are capacity and vulnerability.

Network link capacity is purchased at a premium price in ocean surveillance. Dedicated, long-haul channel bandwidth is very expensive, whether it be a land-based or satellite link. There are two approaches to minimizing the capacity requirements: data compression and sensor-level preprocessing. The data compression alternative involves spatial and/or temporal encoding of the outputs of a given sensor system, so that they may be transmitted at a substantially reduced rate, and reconstructed with acceptable fidelity at the receiving node. The advantage of this method is that full data processing flexibility is preserved at the receiving node, and therefore the maximum amount of information may be extracted from the sensor system, particularly in the case of multi-sensor processing.

The sensor-level preprocessing alternative involves a more extensive, and basically irreversible, reduction in the information rate of the sensor through the extraction of certain parameters or features of the output (e.g., detections, classifications, track information). Only this information is transmitted over the link, at extremely low rates relative to the raw data rate. The latter advantage is offset by the reduced flexibility of the received data, i.e., further processing options are restricted by the much-reduced amount of information available. Also, the individual sensor systems will tend to be more complex due to their processing capabilities, and therefore more costly. It is likely that both of the alternatives that have been discussed would be incorporated into various portions of the ocean surveillance network.

Network vulnerability includes considerations of data encryption, link jamming resistance, and error correction capabilities. Data encryption of both sensor outputs, as well as command and control information, is important to the security and effectiveness of the surveillance network. Depending on the political situation in some cases, covert transmission using spread-spectrum techniques may be required. These techniques may also be necessary to ensure reliable transmission from deployed sensor systems that are susceptible to jamming tactics by the enemy. Finally, the noisy characteristics of certain links, such as telephone lines, may dictate the use of error correction and control methods, so that data "drop-outs" can be avoided.

In addition to considering the technology available for network links, several issues regarding the data communications network as a whole must be considered. The design of a sensor and control data network is very sensitive to the particular solution used for the general OT^2 problem. For instance, if the solution approach to dealing with sensor data involves primarily gathering data into a centralized fusion site for processing, then the network topology would be a centralized topology--a larger version of what is used for gathering data for the ARC.

However, if issues of survivability enter in--as they should--then other network topologies would be more appropriate. If a satellite network was to be used for the network backbone, then a more desirable topology class would be the star polygon. These topologies have several advantages. Most importantly, enemy action cannot take out a central node--thus destroying the network. Furthermore, this approach is convenient when a worldwide distribution of sources and sinks is required. An economic aspect is that many satellites with identical hardware may be used--and if the satellites are missionreconfigurable by uplinked software, then the network may be quickly (even automatically) adapted to new satellites or satellites taken out of service.

Any network that supports a global distribution of sources and sinks would provide the cross-linking of sensor data and control messages necessary to fully utilize concepts of distributed processing and distributed data bases. However, any network of that scope would clearly have to support multiple missions and a wide range of message traffic with different bandwidths, burst characteristics, and destination distribution (one sink versus many sinks). Several of the more common network management schemes are dedicated circuit, circuit switching, packet switching, and message switching. Each of these methods has advantages for certain classes of message traffic.

There are several areas of investigation appropriate to the network issue. One area is determining the utility of various data network topologies applied specifically to OT^2. Associated with this area is the question of what mix of management technologies (packet switching, circuit switching, etc.) is appropriate for OT^2 communication requirements. If multiple users are involved, how do many users access a common system given the need for security and anti-jamming methods? Since different communications technologies are involved (satellite, land line, microwave, etc.), what effects will data lateness over various links of the network have on a specific user such as a fusion center? This is especially important since, if adaptive routing of messages is used to accommodate traffic jams or link failures (like in the ARPANET), data might have random arrival times as opposed to predictable arrivals. Finally, the performance of the network under attack and crisis scenarios should be studied. A mix of analytic and simulative techniques will probably be necessary to identify network problems due to missing or degraded components.

4.0 DISTRIBUTED INFORMATION PROCESSING

Any solution to the Ocean Tactical Targeting problem will require digital and analog processing techniques at all levels. Examples of digital processing tasks at the lower levels are prefiltering and calibration of data at the sensor level, data compression or encoding of data, and sensor level data analysis. At the higher levels, some tasks are management of data bases at the tactical and fleet level, sensor data integration or fusion, target detection and tracking, threat type/target association, threat prioritization at the tactical level (e.g., incoming missiles, approaching aircraft), and fleet level force/threat evaluation and assignment (e.g., assignment of specific ASW components to an identified submarine threat). One possible approach to the system design is to ship data from all sensor sites to a central evaluation center where all of the processing mentioned above is carried out. Fleet components could be commanded based on evaluation of the input data. There are several reasons why this approach is impractical in general.

First and foremost is that the data rate of some sensor systems is much too high to be rooted from a sensor site to an evaluation center. Some processing is necessary to select a data subset, encode or compress the data to reduce the required transmission bandwidth, or to perform pre-analysis of the data to extract a desired component of the information available in the data--again a bandwidth reducing function. At least at the lowest level, then, certain digital and analog processing tasks must be 'distributed' because of communication bandwidth limitations.

Another reason for distribution of processing functions is survivability and flexibility. As with the data communications problem, a level of redundancy is necessary so that if part of the system fails, the remaining resources will (perhaps at some degraded level) still perform the ocean targeting function. An issue which is closely related to system survivability is that of flexibility. Since many elements of an ocean targeting system would be mobile (weapons systems and sensors in both airborne and ocean-based platforms), a fleet commander might wish to allocate a larger portion of all resources to a particular "hot spot." Physically placing the computational resources in or near the activity site could yield benefits such as improved coverage, finer targeting rsolution and shortened system response time. Thus a commander could 'zoom in' to a particular area of interest for special action.

A third reason for distributing computational resources is the concept of task sharing. Some system tasks may be designed so that they can be processed at several different processing nodes. If processing nodes in one part of the system become overloaded, tasks could be shifted to make use of underutilized processing resources at other nodes. For example, destroyers in support of a carrier could process radar data locally while linking sonar data to the carrier where, presumably, a larger processing resource would reside. There are, of course, problems with doing this in general. The first problem is that some processing resources in the system have unique structures and have permanent specializations (e.g., beam forming hardware for acoustic data sources--this equipment can be 'retasked' only in the sense of sensor allocation to specific geographic targets).

A second problem associated with task sharing is the volume of internode communication traffic required to completely shift the processing may be impractical. A tactical data base could not be transferred from one ship to another with surplus processing capability because the lateness associated with the arrival of the complete data base might be comparable to the flight time of an incoming missile.

However, there are many areas in which this could be a valuable concept. One (admittedly crude) example is to partition a surveillance area into cells and assign a data integration center to each cell. In a manner similar to the way aviation traffic is handled by air traffic control centers, sensor data for one geographic cell is handled by one processing center. As targets leave a cell, they are "handed off" to another center. This "cellular" approach has the advantage that surveillance data flow and data processing is more evenly distributed. This approach also has advantages in flexibility. Threat prioritization and resolution can be handled automatically or 'hands-on' as suits the command for a region and the resources available to it. In addition, if it were possible that all centers could, at their option, tap into data sources for other cells (by fan-out within the sensor data network), then multiple centers could focus on one cell for higher system performance in that cell.

The accessibility of a wide range of sensor data to a large number of integration centers has certain implications about the sensor data network associated with an ocean targeting system. An information hierarchy is certainly implied in that there is a wide mismatch of data rates, accuracy, and coverage amongst sensors. Existing operational sensor systems are handled for the most part by individual baronies: an IR satellite, for example, downlinks to its own earth station and its own specialized signal processing hardware. Therefore, a hierarchy is necessary to accommodate existing resources. One way to provide additional crosslinking between sensors and computational resources could be via communication satellites with multiple spot beam antennas for both ground (or air) based sensors and users. Intermediate processing for, say, an IR sensor could be accomplished on the sensor platform or at a ground based station. Whatever results of the processing that the command of the individual barony wished to disseminate could be crosslinked to other sensor net nodes.

As will be discussed in Section 8, there is a need for data bases and processing at the force element level as well as at the fleet level. Many sources of information may be fused to form a fleet level picture. Some information, such as from wide coverage sensors (satellite based staring mosaics, OTH radar, etc.) can also be used within a particular tactical region (or 'cell') for sensor system calibration. An aircraft dropping sonobuoys can maintain sonobuoy position relative to the aircraft by local processing, but needs at least inertial navigation input to maintain position relative to other units and, in particular, weapons systems. The aircraft knowledge of position could be improved if its local processor could tap a satellite downlink for its own position. A ground based acoustic sensor system could calibrate its own coordinate system with the fleet coordinate system if it could tap the location of some target viewed by some other sensor system (OTH radar, etc.) as well as its own. Exchange of data and data processing results by cross-linked satellites (involving, for example, FLTSATCOM) and by a network such as AUTODIN II could open up a host of distributed data base and distributed processing possibilities for an ocean tactical targeting system.

There are several difficult areas that must be addressed to solve the processing problem:

- How are the processing problems or functions to be partitioned into tasks so as to take advantage of multiple processing nodes?
- How can task elements be automatically identified and assigned, and how are task results coordinated?
- If the allocation of tasks is to be distributed, how do various processing resources negotiate for task assignment?
- What intermediate representation of sensor data allows effective utilization by a wider range of processing functions within the targeting system?
- In what ways can existing sensor baronies be interfaced to a network in such a way that their output is more generally useable and at the same time will not interfere with the primary mission of the sensor?

- How can computer hardware and software systems be designed to support a primary function specialization as well as a wide range of temporary specializations?
- If it is the case that it is not practical for all processing systems to have access to all data (e.g., all sensors track file from section 2.0), which is the probable case, which sensor systems should have crosslinks in order to minimize duplicated processing?
- What new computer technology could be developed to handle the extreme response time requirements for threat assessment/threat prioritization/weapon allocation problem in the tactical theatre? An example is STARAN type associative array processors with appropriately configured data bases.

5.0 DATA BASE PROBLEMS

When considering the data base management systems to be used in an OT^2 environment with many sensors, it is essential to approach the problem at the same conceptual level as one approaches the rest of the OT^2 problem. An overall system design should be developed that will integrate the various data types that will be used as input. The design should take into account the relative priorities of different data bases and the similarities and differences in their content and structure. The content and format of data bases are subject to change, and if the management system were too closely tied to specific data base configurations, any change in configuration would require extensive revisions to the data base management system as well.

In the lower levels of the OT^2 hierarchy, there are several requirements for the handling of data in its raw and near-raw state. TARF data must be filed in its raw form upon receipt, but error correcting and field validation processing might also imply a parallel TARF data base with 'corrected' entries. Time series data; weather satellite data, and OTH radar data all require data bases designed for the particular data rate characteristics, burstiness, and access methods needed for the data involved. Because the data bandwidth of sensor data is larger near the sensor, more attention must be paid to sensor data base performance.

Since a data base maps data flow from a source process to a sink process (as well as acting as a repository), the access pattern or 'key' used to store the data is often different from the retrieval pattern. For this reason, access to a given data base should not be by a single key or by a hierarchically-ordered set of keys. Such an access plan assumes that requests to the file for information will always be of the same kind (i.e., by STAR target number), and is incapable of efficient retrieval by lower-ordered keys. A number of data access methods exist which handle multiple keys efficiently.

These methods can be tuned to give optimal retrieval time for the most commonly used keys without sacrificing general efficiency no matter what key or combination of keys is specified.

One such access method is partial match retrieval using hashing and trie algorithms. This is an associative retrieval method with proven upper bounds on the number of disk accesses required to retrieve information specified with varying numbers of keys.

In all hashing algorithms, a function is applied to the original set of key values to create a new value which is the address in the file where the item will be stored. To retrieve the item, the key is hashed with the same function to produce the item's address. The unique aspect of partial match algorithms is that they can retrieve data items which have specific values for some keys and unspecified or wild card values for other keys.

A system of this kind has very desirable properties for a particular application like an OT^2 test bed. Insertion of new data is very fast. The system permits access to data bases on an assortment of keys and facilitates retrieval of groups of data items. For example, in dealing with input data, an operator might wish to confine attention to peaks above a stated coherence level whose time of receipt was within a certain time slot, or peaks from a given pair of sensors, or peaks from a given geographical part of the ocean that satisfy various fequency and water time constraints. If this data is organized by a single key or by a hierarchically-ordered set of keys, few queries of this kind can be handled efficiently, it at all. A multiple-key, partial-match hashing algorithm handles queries of this nature easily. When the software receives a retrieval request, specified key values are left intact and values for unspecified keys are treated as wild cards.

Another multiple key access method employs a Normal Multiplication Table as an index to the data base. This method is well-suited to data bases where the number of data items is large relative to the number of distinct key values.

As was the case with trie algorithms, system performance can be tuned by ranking the keys according to expected fequency of use, singly or in combination. Perhaps the greatest difference between the two methods is that the Normal Multiplication Table is constructed from the contents of a specific data base while the trie algorithm is designed to handle any data set that contains items that can be described by the current set of keys. This difference has important implications for an OT^2 test bed. The Normal Multiplication Table method would work well for artificial input data or for input data that arrived in discrete blocks, as is the case with the "epochs" currently being used in experiments at the ARC. The algorithms that build the table are relatively simple and fast, and once the table is built, it would provide extremely fast information retrieval.

Because they are attribute-based, both of these methods adapt easily to changes in file format or content. They require far less core space than the standard inverted indices or multilist directories. If proper data definition is done at the outset, these methods can be used across files and therefore can link together data from different input data bases.

Information dealt with at higher levels in an OT^2 system has been extracted from data at the sensor by various processing techniques. Higher level OT^2 data bases will contain such information as target identification, position, course, and speed. Target identification may include hull type, known weaponry, intent, and severity of threat if an enemy. In addition to representing the current state of the surveillance area, a high level data base may contain a target history of past movements and events, and maintain a list of sensors contributing to a piece of high level information.

A commander may need to relate the positions of certain fleet elements to the sea condition for that particular area. However, the location of a data base containing weather may be physically separated from the command center data base. Therefore, issues concerning distributed data bases need to be addressed. Various fusion centers will maintain data bases relevant to their own sensor suite. However, decisions made at a central location require that all these data bases be accessible in a uniform manner. In addition, procedures will be required for the event of the failure of a remote data base.

The wide range of types and formats of data required at the fleet level may argue for a relational data base design. Ultimately, one would wish to automate fairly complex queries of the data base to identify particularly important situations without human intervention. The structure of a relational data base is well suited to the associative searches that are required to handle queries of this kind. Because the information is organized by its attributes and does not have a structure imposed on it, it can be efficiently searched for records which match a combination of key values that characterize a particular threat or other tactical situation. In instances where speed is critical, special hardware, such as fast array processors, can be used.

For the highest levels in an OT^2 system, relational/distributed data base technologies exist that can handle the sorts of information required by a commander at the fleet level. Some appropriate work with human interface technology and relational data bases has been accomplished through the ACCAT project. However, serious issues of system performance have to be addressed. Also, it is likely that several data base access techniques, some of which were discussed above, would be more suited for data at the local tactical level because of response time constraints. The nature of the data being dealt with at levels closer to the sensor is also quite different. The point is that though the interfaces to different data base systems should be similar, the interfaces should be independent of the access methods used. The access methods should depend on the type of data handled, types of access, and response time requirements. A uniform interface, on the other hand, will lend itself to uniform representation of sensor information and thus more flexible utilization of data throughout the targeting system.

Thus, several problem areas that must be addressed to properly use data base technology in an OT^2 system are:

- Identification of what types of data bases are actually required for the OT^2 problem. Examples range from acoustic data bases at an acoustic data processing center to data bases containing hull number and ship captain for a particular ship at the fleet level.

- What data base access techniques are optimal for data at various levels in the hierarchy?

- Can specialized hardware be applied to certain time-critical applications? Cellular-logic devices, head-per-track relational data base machines, associative array processors, and mass storage devices utilizing content addressable blocks are examples.

- At what level in the system are distributed data bases appropriate? What types of information should be distributed and what types tend to be centralized? How can consistency between such data bases be maintained? How is the problem of failure of remote sections of the data base handled (fail soft)?

- For time-dependent data, what techniques may be used to identify and delete unimportant data from the data base in order to avoid data base overflow?

6.0 MEASUREMENT FUSION

In considering the fusion problem embedded in the OT^2 problem, one must give paramount importance to the data available and its utility. In concept formulation one should not be tied too closely to any given set of sensors or sensor system but rather to concepts and potential suites of sensors. There are a large number of possible sensors whose data could be potentially useful in the fusion center. Types of sensors that could be considered are:

- Acoustic;
- Electro-magnetic;
- Optical;
- Infra-red;
- Thermal; and
- Human sources.

In most cases both passive and active sensors could be considered as well as both fixed and mobile locations and many possible sensor placements (satellite, land, air or sea).

With this plethora of possible data and data types many questions and problems arise. In the simple case of detecting one signal in noise or of tracking a given target (with data for that target) the addition of more data either helps or does not hurt the processor. This is not true of the data association problem, particularly with limited communication, computational or human resources. It is possible in the multi-target/sensor arena to turn a situation, where one could detect, track and classify a number of targets, into one where this is impossible. This could be made to occur for any given system by the addition of more clutter, more real targets, or even more good data from existing targets. This could occur because of the clogging of limited communication channels, the creation of too many possible or feasible combinations of data in a multi-target tracker or saturation of computational resources. In this somewhat novel situation it becomes very important to define the quality measure that one will use to evaluate the eventual system output. It must be noted that there could be multiple (possibly conflicting) requirements on this system requiring a multi-dimensional quality measure.

Since one must finally operate the data association, detection, classification, localization and tracking systems on a set of data from a set of sensor types, one can think of each possible selection of data as one set of possible or available data. In evaluating a given set of some allowable communication/computer resource suites, one could choose a number of quality measures. One could perhaps, from a technical point of view, define the information content of a set of measurements--some kind of entropy measure on an absolute basis. A measure of a set's ability to give high resolution target tracks (most precise tracks) could be one such quality measure. Another would be a measure of the ability of a set to allow separation of different targets or different classes of targets. In this case one could look at the features that one wanted to separate, define a distance measure and use pattern recognition techniques. One might opt for the set that would allow the largest number of targets (possibly of a single class) to be detected while minimizing the probability of false alarm.

It should be clear that if a system used its resources to track, detect or classify loud targets it might not have the resources required to track quieter or other particularly important targets. The measurement sets allowing best detection might not even be observable sets for track reconstruction or target classification or separation (data association).

Other means of measurement quality could include time lateness, independence of other measurements in the set, and correlatability with the measurements from different systems. Even if some measure of goodness data from one sensor system is judged to be the best, additional data from that system on a given target would not be independent. Thus, it might be better to include data (even of a lower quality) for a different sensor system to confirm the existence of a given target.

Major research problems exist in the data fusion area. Some of these are data independent but many others depend on the details of the sensor system, the statistical characteristic of the measurement systems and even cost of data transmission, computation and other factors, allowing a reasonable comparison of resources on a cost or uniform basis.

7.0 MULTI-SENSOR/MULTI-TARGET TRACKING, DETECTION AND CLASSIFICATION PROBLEMS

The multi-sensor/multi-target tracking/detection/classification problem area is a formidable one. For a given surveillance area, there may be a large number of surface/subsurface ships, aircraft, and other acoustical signal sources in close proximity to one another such that one is confronted with a dense multi-target environment.

To illustrate the complexity of the problem, one could first consider the case where there is one sensor system that is receiving signals from several targets as illustrated in Figure 6 below, where each of the targets could potentially represent aircraft, surface/subsurface ships, etc. The sensed signal might even be caused by sea-induced noise.

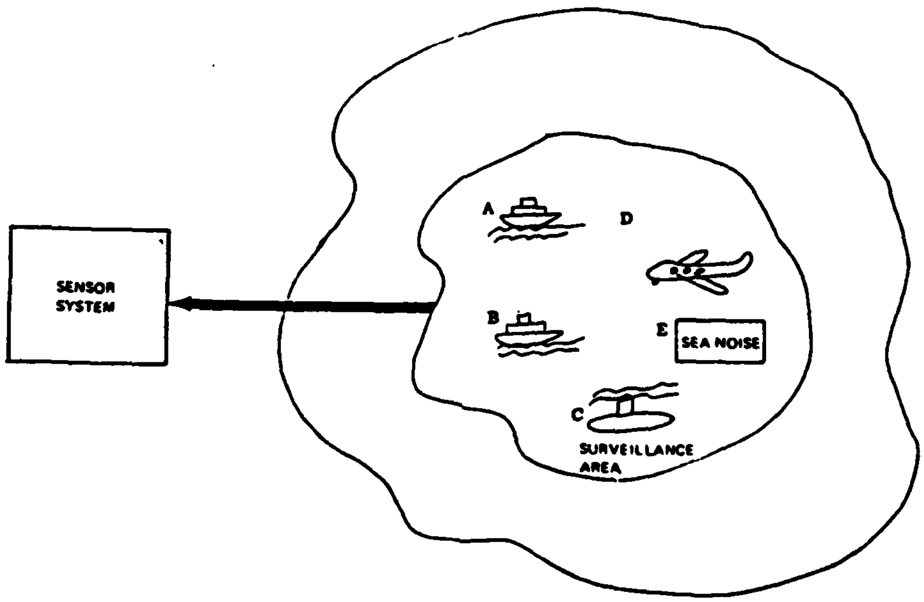


Figure 6. The multi-target problem.

This problem is defined as the single-sensor, multi-target problem. When the set of targets (A,B,C,D) are operating in close quarters and/or when the sensed signals have a low signal-to-noise ratio, a fundamental problem is that of data association (return-to-track). That is, for each incoming measurement, how do we attach it to the correct target that produced this measurement, or is the measurement caused solely by noise (false alarm or clutter) as in box E in the above diagram. This is fundamentally a target detection problem. There have been a number of approaches used in the past in solving this problem, and they have met with varying degrees of success. They range in scope from very simple procedures such as manual selection of data by a human operator to more computationally sophisticated methods that can automatically handle the data association problem without operator intervention.

The manual procedures involve careful preprocessing of the data by the operator, and although laborious and time-consuming, have met with some success on real data. They work well when the targets are widely separated and the signals have a high signal-to-noise ratio, but in a "dense" environment and/or low signal-to-noise ratios, these approaches are not very suitable. It is the latter set of automated algorithms that fare better under this type of scenario, and lately have been shown to exhibit quite good tracking performance on both real and simulated data. Graphic examples of this were found during the SASE experiment.

For the most part, all of the automated methods employ extended Kalman filtering of one form or another to perform data association. This is done using statistics generated by the filter and subsequently comparing them to a set of specified thresholds. In addition to the data association task, the filter is structured to do the target tracking in parallel so as to provide target estimates in a preferred coordinate system (for instance, position, speed and course).

In addition to the problems already noted in the above discussion, one must now face the issue of correlating the measurements from different sensor systems for the same given target. This is a multi-sensor measurement fusion problem and a few solution approaches to it have already been suggested; namely, independent multi-target tracking for each sensor system, and then the combination of the individual sensor tracks into a resultant set of target tracks.

To illustrate the complexities involved in going to a more elegant approach that would be an improvement to the above, it might be helpful to pose a specific approach as an example. This approach would have the tendency to increase the dimensionality of the observation space in a sequential fashion thereby increasing the information content of the data set. The key features can best be described with the aid of the block diagram in Figure 8. Essentially it involves implementing a multi-target tracking filter to perform the detection, classification, and tracking functions for each of the individual sensor systems A to Z, respectively. The tracks generated by each of the filters are then combined into a composite set of tracks. If a given target is being sensed by more than one sensor system, the filter for one sensor system should generate a target track that is "reasonably close" to a target track generated by the filter for another sensor system. The major objective here is to correlate those target tracks of different sensor systems that correspond to the same target. This is the function of the track clustering block shown in Figure 8.

A variety of clustering approaches can be used to identify tracks belonging to a common target. They range in complexity from simple procedures such as a least squares fit to more complex procedures that use the confidences associated with a set of target tracks to "weight" the results of the clustering approach. This latter approach can be implemented recursively using an extended Kalman filter which has as input measurements the target track

When using the filter for data association and target tracking, care must be taken in selecting a filter model that accurately reflects the physics of the actual target motion. The usual assumption is constant rhumb line or great circle motion. This is normally adequate unless the target starts maneuvering. When this occurs, the filter could misinterpret measurements from a maneuvering target for clutter when performing the data association task, and not use the measurement at all in forming a target track. For abrupt maneuvers, solutions to this problem can normally be achieved by artificially increasing the filter's process noise covariance matrix. However, for more subtle maneuvers an obviously more sophisticated algorithm must be applied. Several innovative approaches have been proposed recently to differentiate between maneuvers of this sort and clutter. One of the more promising of these appears to be the accumulation of frequency sensitive measurement residuals as a test statistic for maneuver detection.

The above discussion has thus far only considered a single sensor system. A much more complex problem is introduced by considering the multi-sensor system, multi-target problem depicted in Figure 7 below.

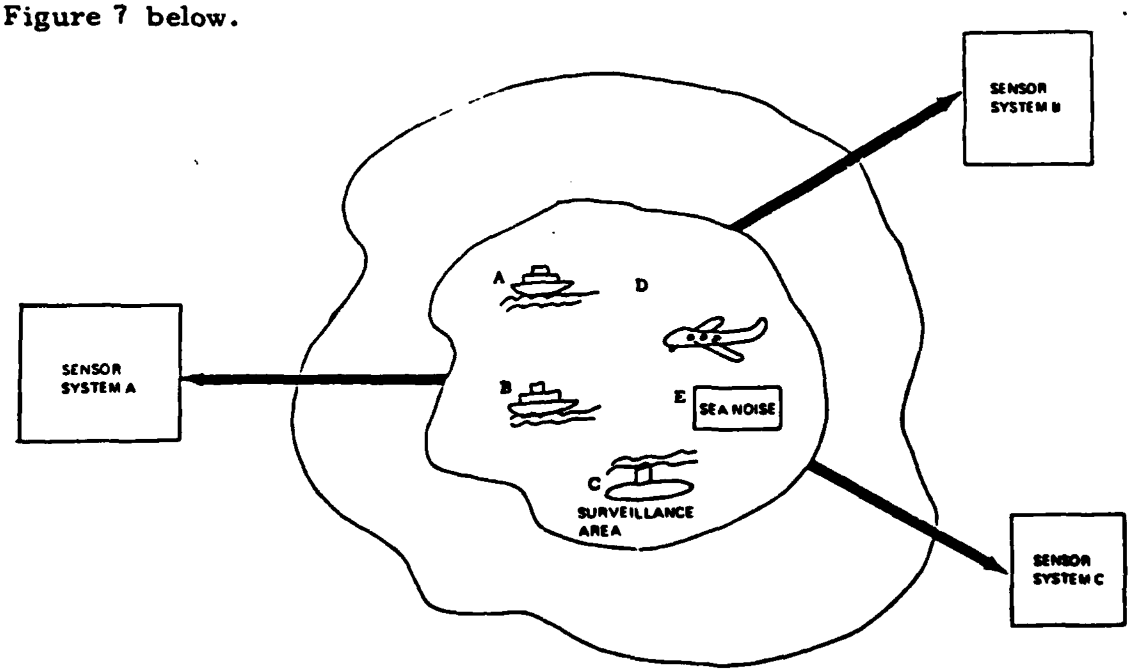


Figure 7. The multi-sensor/multi-target problem.

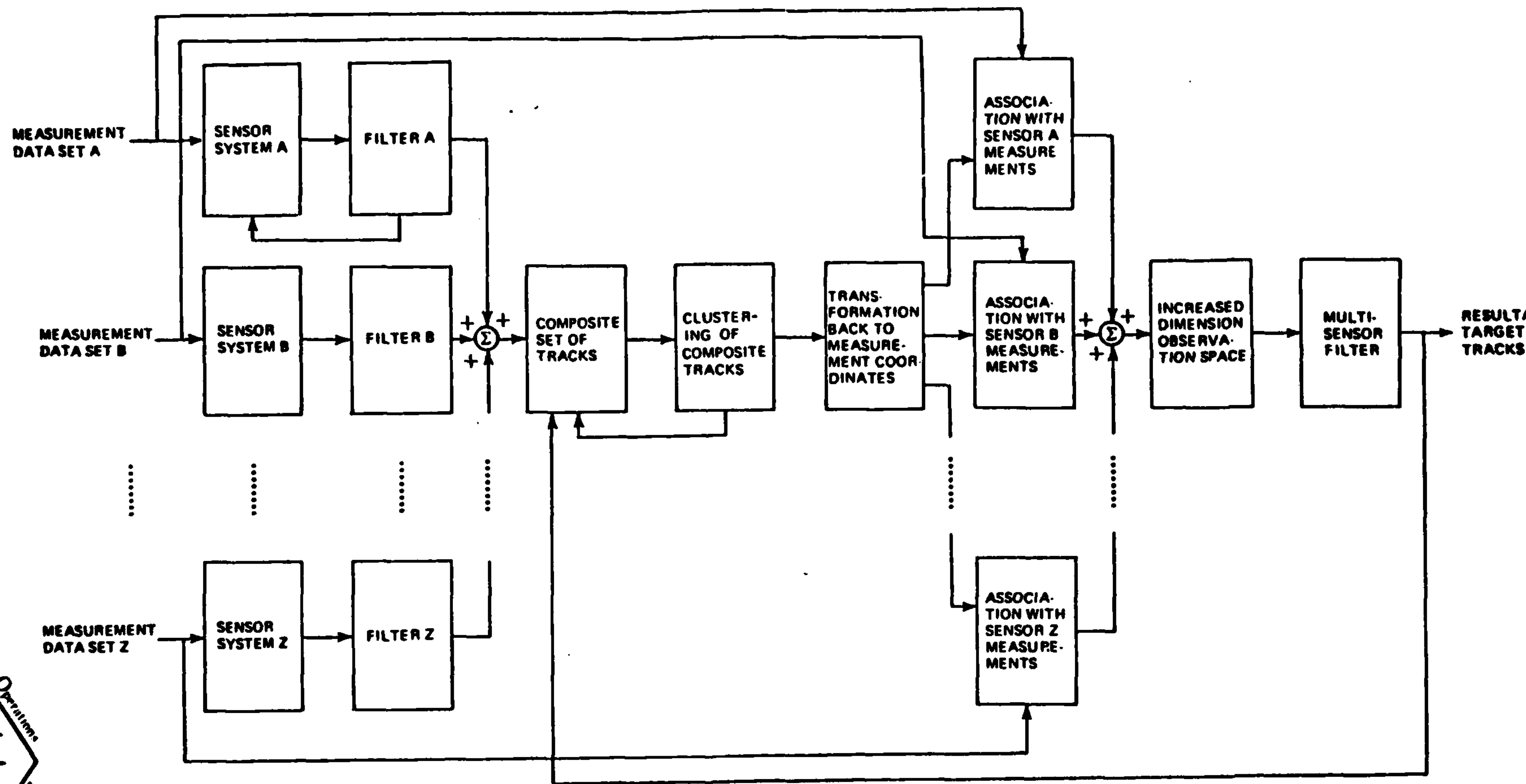


Figure 8. Flowchart of potential sensor fusion structure.

estimates and covariance matrices generated by the filters for each of the sensor systems A to Z, respectively. This comprises the inner feedback loop of the track clustering algorithm shown in Figure 8.

Having obtained a set of clustered tracks, they can then be transformed back into a set of estimates in measurement coordinate units for each of the sensor systems. Association of the actual measurements with the measurement estimates above is obtained by identifying those measurements lying "close" to the estimates for a given target track from the cluster for each of the sensor systems.

The measurements for the given target track are then correlated, i.e., they are identified as having come from the same target, and are combined into an increased dimension observation space having more information content than the measurements for each of the sensor systems by themselves. A filter can then be designed to process this multi-sensor data to produce a more accurate target track. This procedure continues until all member tracks of the target track cluster have been processed.

At this point, the outer feedback loop on the track clustering algorithm can be closed to produce a refined set of target tracks. This is done by repeating the track clustering, measurement data association, and multi-sensor filtering. The only difference here is that the track clustering is initiated on a set of composite tracks that were generated by the multi-sensor filter. It is easy to see how this can be put in an iterative mode, i.e., continue to feed back the resultant target tracks into the track clustering algorithm, until no further change in the resultant target tracks is noted.

Again, this approach has been intended as an example and not as a specific solution to the multi-sensor measurement fusion problem. The purpose here is to illustrate the intricacies associated with any approach that would extract all of the information embedded in a set of measurements from different sensor systems, and use it in an optimal fashion to enhance our knowledge in the areas of detection, classification, localization and tracking of target threats within the surveillance area.

8.0 FORCE AND WEAPON ALLOCATION

The ultimate goal of an Ocean Tactical Targeting System is to bring the proper weapons to bear on the targets in order to neutralize them while minimizing the losses among our own forces. There are three issues which need to be addressed in this area:

- Threat prioritization;
- Friendly force/target assignments; and
- Real-time battle management.

As part of the threat prioritization function, the data base management system will provide the information to allow priorities to be assigned to each of the threat. This prioritization will be assigned based on:

- Threat type;
- Location; and
- Intent (e.g., direction of movement).

For example, a high priority will be given to an enemy ship carrying cruise missiles near a friendly land mass, whereas a low priority might be assigned to an enemy ship carrying only surface-to-air missiles located far from any friendly ships or land masses.

In the actual targeting/weapon allocation problem there are hierarchical levels of command and control which must be considered for effective use of the available weapon resources. The top level of this structure (see Figure 9) will allocate groups of threats to friendly forces based on the threat priorities. As time evolves, the threats will move, their assigned priorities will change, and the force/target allocations will be modified. These changes will take place relatively slowly so that these assignments can be made in an optimal manner.

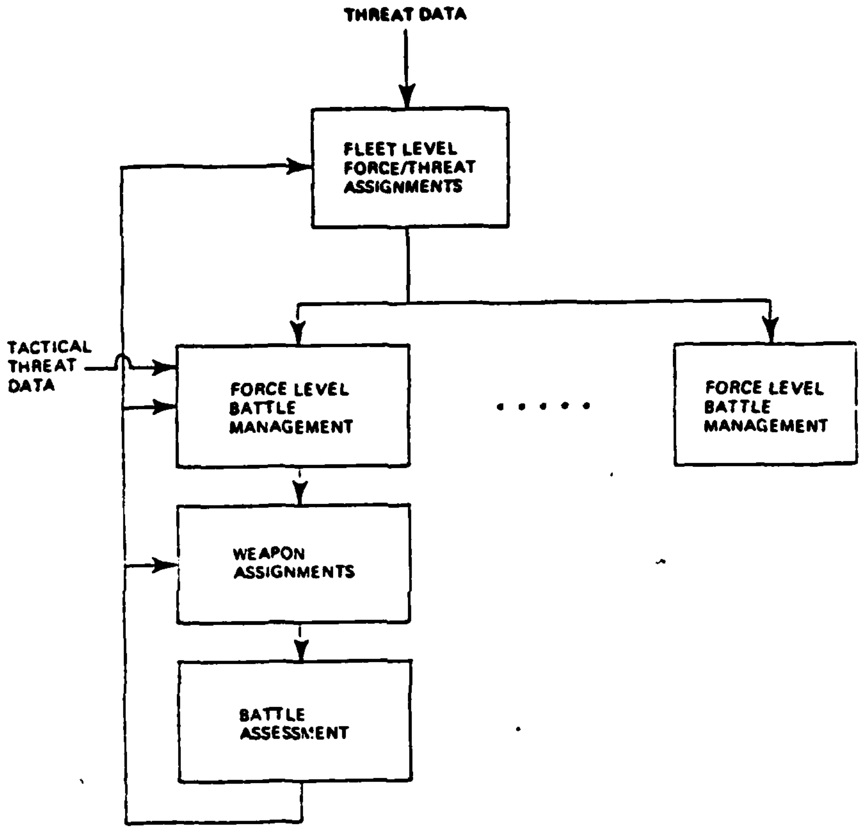


Figure 9. Force and weapon allocation.

The force level battle management function takes place at a lower level. This includes real-time ship or weapon assignment to threats, and real-time reallocation of forces/weapons based on battle assessment data. This level is further complicated by the need to counter enemy attacks, as well as to destroy the assigned threats. Because of the real-time requirement of this battle management function, some degree of optimality in weapon allocation must be sacrificed to achieve the real time capability.

Such a two-level command and control structure allows optimality of force/threat allocations to be emphasized because of the relatively slow dynamics at this upper level while delegating to a lower level the real-time battle management function.

This is a complicated resource allocation problem and there are a number of specific issues which need to be addressed:

- What is the best division of functions between the two levels?
- How detailed must the weapon allocation be, e.g., ship level or weapon level?
- How are the offensive and defensive issues of battle management optimally addressed?
- How can these approaches be automated, and can the concepts of artificial intelligence be used?

This might allow real-time planning to improve battle management at a lower level, freeing the major human decision-makers to make major policy decisions or system overrides.

9.0 FINAL TARGETING

When the long range surveillance system has detected, tracked and classified a target to the best of its ability, one generally will still not know the target position well enough to launch a weapon system against it. It may also be politically unacceptable to launch an over-the-horizon weapon without a final classification system that will confirm the target classification made at long range. There are many unsolved research problems inherent in end point targeting and classification. The first targeting could be done using a deployable sensor (perhaps a sonobuoy field) in conjunction with one or more aircraft and a weapon system.

A strike system designed to destroy a submarine must obtain positional information about the submarine relative to the strike aircraft. In order to ascertain this information, a sonobuoy field may be used. The field must contain enough buoys to obtain sufficient observables for determination of the necessary position and velocity information. The optimum number is scenario dependent. The designation aircraft which accomplishes the target function must derive the information from the observables and transfer this information to the strike aircraft which must, in turn, establish an attack position and launch its weapon. The type of weapon guidance used is also scenario dependent. The weapon must enter the water near enough to the submarine to allow the terminal sensor to find and lock onto it. A stealthy attack is required in order to negate the target's ability to employ countermeasures or counterthreats and to deny performance of its main function. Thus, a detailed analysis of the scenarios, function and operational capability of such a sonobuoy field target strike system must be performed. From earlier ORINCON studies performed for NADC, it appears that there may be fundamental observability problems which must be investigated. Problem areas include the strike scenario, the prediction of target position and velocity, the weapon guidance modes, transfer alignment and the effect of the terminal sensor.

The problem of utilization of a sonobuoy field to obtain targeting information and to perform a strike mission against a submarine using guided weapons requires a detailed consideration of system options and their accuracies. Targeting utilizes the sonobuoy reference system to obtain the position and velocity of the submarine relative to the aircraft. This includes the aircraft Doppler and inertial navigation systems, and an azimuthal measurement. The interferometric angular determination requires an antenna pair baseline on the aircraft and a transmitter on the sonobuoy. The targeting of the submarine has uncertainties involved due to the aircraft inertial system and interferometer errors. These cause a degradation of the submarine location and velocity relative to the designation aircraft. It has been found that with the use of sonobuoys this designation problem is highly sensitive to aircraft maneuvers.

When the designation aircraft is not the strike aircraft, further problems occur. In particular, the strike aircraft position relative to inertial space is in error due to its inertial navigation system errors. Thus, since the designation aircraft position relative to inertial space also contains uncertainties, the location of the strike aicraft relative to the designation aircraft is in error. This implies an additional uncertainty in the location and velocity of the submarine relative to the strike aircraft.

The performance of the strike system may be affected in various ways. Of course, the basic intent and, thus, the ultimate performance measure is that of destroying the submarine. In order to accomplish this, the weapon must be placed within a certain volume about the submarine in order to assure that the terminal sensor can activate and lock onto the target. Failure to place the weapon within this boundary will result in an unsuccessful strike, and is due to the system errors discussed. These errors are dependent upon strike parameters such as launch range, type of guidance laws and systems, number of sonobuoys in the field, proximity of strike aircraft to designation aircraft and ability of the designation aircraft to maneuver to obtain the proper observability.

Another performance measure is that of the ability of the aircraft to survive any counterthreat. This may require a "launch and run" scenario rather than an "on-station until destroy" scenario. The ability of the strike system to fire at long range is certainly important in the counterthreat environment. Other measures include the ability to distinguish real from imaginary threats, to acquire, designate and strike quickly and reliably.