(9) Progress rept. no. 3, Quarterly rept. no. 2

MOST Project-3

THIRD PROGRESS REPORT — SECOND QUARTERLY REPORT

(11) 15 January 1970

(12) 23 p.

(15) (10) D. W. Winfield

General Electric

Contract No.: N00024-69-C-1268

Title: (6) Multiping Data Processor for Target Detection

During the previous quarter, the multiping processor design was finalized and parameters selected for the detection and tracking portions of the multiping processing algorithm. Work also progressed and is continuing on obtaining quantitative performance results on the multiping processor through Monte-Carlo techniques. Preliminary performance results will be obtained by January 30, 1970. Figure 1 shows the block diagram of the multiping receiver configured with an output display and automatic detection device. An analysis of average total costs for multiping sonar decisions was made and will be useful in comparing competing decision algorithms. This analysis is presented in the attached Technical Memo No. 1. Technical Memo No. 2 develops the Kalman filter equations and describes the adaptation of the Kalman filter to our specific track detection problem.

During the next quarter quantitative performance results of the multiping processor will be obtained via Monte-Carlo methods using the digital tapes generated containing noise and signal plus noise with time and frequency spreading of arriving target associated paths. This data was generated using the target/environment model described in the Second Progress Report dated October 15, 1969.

(14) TM-1, TM-2

149 510

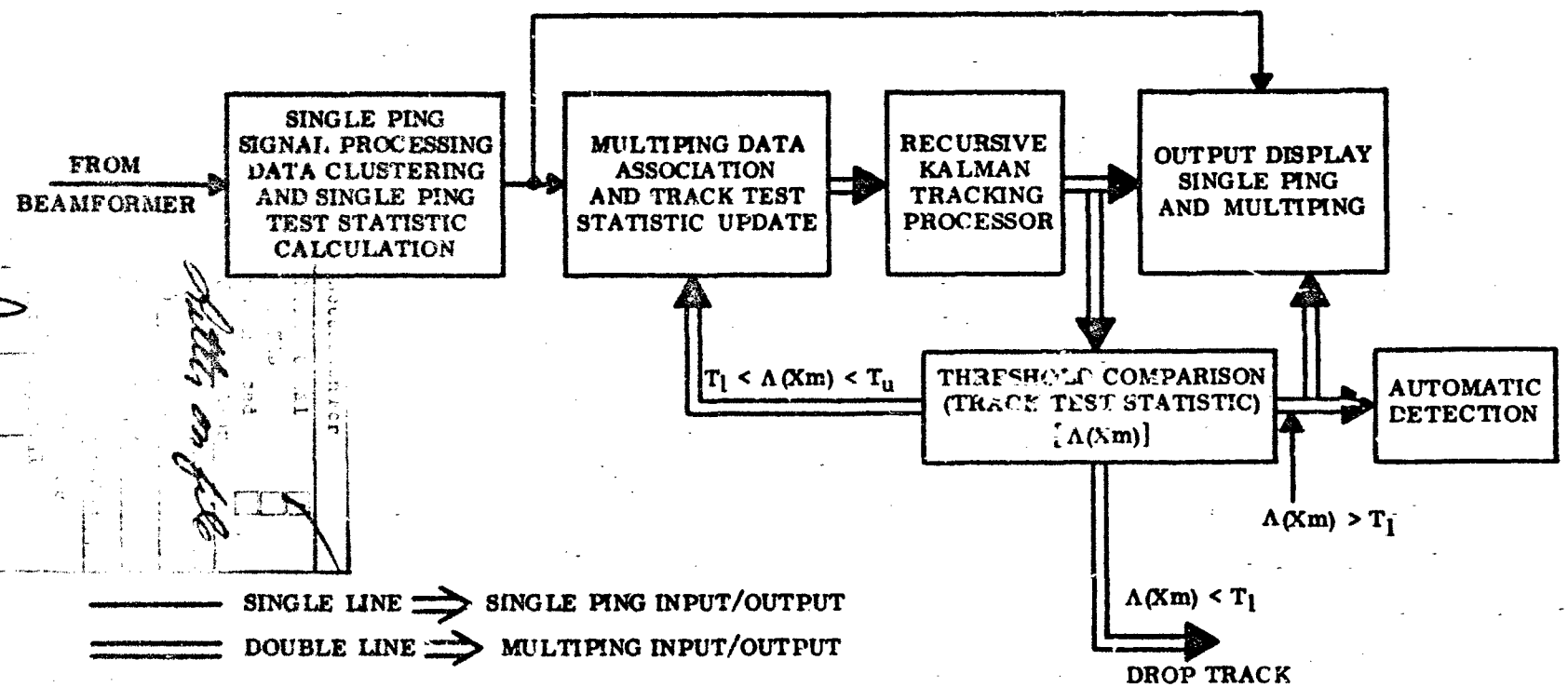

Figure 1. Multiping Receiver Block Diagram

Technical Memo No. 1

January 2, 1970

T. G. Kincaid

Contract No.: N00024-69-C-1268

Title: Average Total Costs for Multiping Sonar Decisions

1. INTRODUCTION

This report is a discussion of the average total cost of making a "target present" or "target absent" decision from the information contained in a sequence of sonar pings. This cost is useful for comparing competing decision algorithms. The total cost is computed as a sum of (1) the costs of terminal decisions, and (2) the cost of pinging.

The discussion is divided into two parts. The first part is the derivation of a formula for the total cost of using any sequential decision algorithm in a multiping sonar receiver. The derivation presented here is a straightforward extension of existing cost computing procedures for one possible target track to the case of many possible target tracks. The second part is a description of three sequential decision algorithms, and a comparison of their merits based on costs, error probabilities, and ease of implementation.

2. COMPUTATION OF AVERAGE TOTAL COSTS

The multiping sonar receiver makes its "target present" or "target absent" decisions on the basis of the returns from a sequence of sonar pings. We assume a receiver which examines each possible target track after every ping, and chooses between three alternatives for each track: (1) decide "target present" in the track, (2) decide "target absent" in the track, or (3) make no decision and ping again.

To evaluate this receiver, we consider the following experiment. The sonar system continues to ping until a decision of "target present" or "target absent" is made for every track. The receiver is then evaluated on the basis of the average total cost of performing this experiment.

In order to derive a formula for this average total cost, we introduce the following notation.

k = the track number

k_{max} = the total number of tracks

I_k = a random variable which takes on the value 1 when the decision is "target present" in track k; and 0 when the decision is "target absent" in track k

j_k = a random variable which takes on the value 1 when a target is actually present in track k; and 0 when a target is actually absent in track k

$c_d(i_k j_k)$ = the cost associated with the experimental outcome $i_k j_k$

c_f = the fixed cost of a ping, and is assumed the same for each ping (this cost would include things like the cost of alerting the target and the energy required to generate the ping)

c_p = the cost of processing a track for each ping, and is assumed the same for each track and each ping (unlike c_f, this cost disappears if a decision is made which terminates a track)

c = a random variable whose value is the total cost of the experiment

$p(i_k j_k)$ = the a priori probability of the event $i_k j_k$

n_k = a random variable whose value is the number of pings required to make a decision in track k

n = a random variable whose value is the number of pings required to perform the experiment, i. e., to make a decision in all tracks (n is the largest value of n_k at the end of the experiment)

The average total cost of the experiment is the sum over all tracks of the average decision cost for each track, plus the cost of pinging required to perform the experiment. The average decision cost is

$$\langle \text{decision cost} \rangle = \sum_{k=1}^{k_{max}} \sum_{j=0}^{1} \sum_{i=0}^{1} c(i_k j_k)\, p(i_k j_k) \tag{1}$$

The average pinging cost is

$$\langle \text{pinging cost} \rangle = \langle c_f n + \sum_{k=1}^{k_{max}} c_p n_k \rangle = c_f \langle n \rangle + \sum_{k=1}^{k_{max}} c_p \langle n_k \rangle \tag{2}$$

The average total cost of the experiment is therefore given by

$$\langle c \rangle = \sum_{k=1}^{k_{max}} \left[\sum_{j_k=0}^{1} \sum_{i_k=0}^{1} c(i_k j_k)\, p(i_k j_k) + c_p \langle n_k \rangle \right] = \sum_{k=1}^{k_{max}} \langle c_k \rangle + c_f \langle n \rangle \tag{3}$$

where

$\langle c_k \rangle$ = the average track cost

This result shows that we must consider the average track cost $\langle c_k \rangle$, plus the total cost of pinging, when comparing the cost of various multiping algorithms.

The expression for the track cost can be further broken down by noting that

$$\langle n_k \rangle = \sum_{n_k} n_k \, p(n_k)$$

$$= \sum_{j_k} \sum_{n_k} n_k \, p\,[n_k/j_k]\,(j_k)$$

$$= \sum_{j} \langle n_k(j_k) \rangle \; p(j_k) \tag{4}$$

Therefore

$$\langle c_k \rangle = \sum_{j_k=0}^{1} \left[\sum_{i_k=0}^{1} c_d(i_k j_k) \, p\,[i_k/j_k] + c_p \, \langle n_k(j_k) \rangle \right] p(j_k) \tag{5}$$

In order to study the average track cost for a single track, we drop the subscript k and define the following.

$p\,[i_k = 1/j_k = 0]$ = probability of a false alarm = α

$p[i_k = 0/j_k = 1]$ = probability of a false dismissal = β

$p[i_k = 1/j_k = 1]$ = probability of a true alarm (detection) = $1 - \alpha$

$p[i_k = 0 / j_k = 0]$ = probability of a true dismissal = $1 - \beta$

$p(j_k = 1)$ = a priori probability of a target present = p

$p(j_k = 0)$ = a priori probability of a target absent = q

Substituting these quantities into $<c_k>$, rearranging terms, and expanding the summations gives

$$<c_k> = c_d(00)\, q + c_d(11)\, p + \left[[(c_d(10) - c_d(00)]\, \alpha + c_p <n(0)> \right] q + \left[[(c_d(01) - c_d(11)]\, \beta + c_p <n(1)> \right] p \qquad (6)$$

We assume the cost of a wrong decision is greater than the cost of a right decision, i. e.,

$$c_d(10) > c_d(00)$$

$$c_d(01) > c_d(11) \qquad (7)$$

Therefore all the terms in Equation (6) are nonnegative.

3. MULTIPING SONAR SEQUENTIAL DETECTION ALGORITHMS

In this section we describe three sequential detection algorithms, and compare their relative merits on the basis of costs, error probabilities, and ease of implementation. The three algorithms are optimum sequential, Wald sequential, and truncated Wald sequential.

Each of these algorithms are developed in the literature considering only one track. Equation (3) shows how to evaluate these algorithms in the multiple track situation.

a. Optimum Sequential

The optimum sequential test is described by Birdsall and Roberts (Ref. 2). This test assumes that a maximum number of pings n_{max} are allowed. Within this constraint, the algorithm minimizes the average track cost $\langle c_k \rangle$. In the multiping sonar case there are usually a large number of tracks and a small maximum number of pings. This combination makes it virtually certain that n_{max} pings will be required to make a decision in every track. Thus $\langle n \rangle = n_{max}$, and the average total cost given by Equation (7) will be minimized if the average track cost is minimized, i. e., if the optimum sequential test is used in each track.

The optimum sequential test compares the likelihood ratio

$$\lambda = \frac{\text{probability of the received sonar echo given a target present}}{\text{probability of the received sonar echo given a target absent}} \tag{8}$$

to two thresholds after each ping. If λ is above the upper threshold, the decision "target present" is made, and the test terminates. If λ is below the lower threshold, the decision "target absent" is made, and the test terminates. If λ is between the thresholds, no decision is made, and the test continues to the next ping. The thresholds move closer to each other on each successive ping, until they coincide on ping n_{max}, which forces a decision and terminates the test. At each stage of the test, the tradeoff is between taking another ping to improve error performance and the cost of pinging again to obtain such an improvement.

The advantages of the optimum sequential test are: (1) the ability to set an upper limit on the number of pings, and (2) the fact that costs are less than any other sequential test with this feature. The disadvantages of the optimum sequential test are: (1) the difficulty of computing the thresholds, which are based on uncertain costs and prior probabilities, and (2) the difficulty in relating threshold settings to posterior error probabilities α and β, which are often the only meaningful performance measures when costs and prior probabilities are unknown.

b. Wald Sequential

The Wald sequential test (Ref. 3) is the special case of the optimum sequential test for which n_{max} is infinite, i.e., there is no limit to the number of pings allowed. Compared to any other test with the same posterior error probabilities α and β, the Wald Sequential test minimizes the average number of pings $\langle n(0) \rangle$ and $\langle n(1) \rangle$ required to make a decision in each track (Ref. 4). This minimization is independent of the assigned prior probabilities p and q, and the costs c_d and c_p. Thus for specified α and β, the Wald sequential test minimizes average track costs. However, the lack of an upper bound on the num-

of pings could be a considerable disadvantage when the algorithm is being used for a large number of tracks. The value of $<n>$ could be very large, and the cost term $c_f <n>$ in Equation (3) might be intolerable.

In the Wald sequential test the likelihood ratio λ given in Equation (1) is compared with two fixed thresholds, and decisions made in the same manner as in the optimum sequential test. These thresholds can be set to achieve any desired posterior error probabilities α and β, or they can be set to minimize average costs.

The advantages of the Wald sequential test are: (1) no other test has fewer average pings to decision per track for the same posterior error probabilities α and β, and (2) the thresholds are simply related to α and β, and remain constant throughout the test. The main disadvantage of the Wald sequential test is the lack of an upper bound on the number of pings, which can be quite costly when there are a large number of tracks, and a high fixed cost of pinging c_f.

c. Truncated Wald Sequential

The truncated Wald sequential test is the same as the Wald sequential test except that an upper limit n_{max} is placed on the number of pings. If a decision is not made by ping number n_{max}, then a single threshold is set and a decision made. The truncated Wald sequential test is an attempt to combine the simplicity of the Wald sequential test with the limited number of pings feature of the optimum test. If the maximum allowed number of pings is large enough, it is reasonable to expect that the performance of the truncated Wald sequential test will be approximately the same as the Wald sequential test.

Comparing the truncated Wald sequential test with the optimum sequential test and the Wald sequential test, the truncated Wald sequential test (1) has a higher average track cost than either of the other tests, (2) has a higher average total cost than the optimum sequential test with the same n_{max}, (3) can be made to have a lower average total cost than the Wald sequential test in most situations where c_f is much higher than c_p, and (4) is simpler to implement than the optimum sequential test.

REFERENCES

1. Bussgang, J. J., and Middleton, D., "Optimum Sequential Detection of Signals in Noise," IRE Trans. Inform. Theory, December 1955.

2. Birdsall, T. G., and Roberts, R. A., "Theory of Signal Detectability: Deferred-Decision Theory," Journ. Acoust. Soc. Am., Vol. 37, No. 6, pp. 1064-1074, June 1965.

3. Wald, A., Sequential Analysis, John Wiley and Sons, Inc., New York, N. Y., 1947.

4. Wald, A., and Wolfowitz, J., "Optimum Character of the Sequential Probability Ratio Test," Am. Math. Stat., Vol. 19, p. 326, 1948.

Technical Memo No. 2

January 2, 1970

D. L. Jordan

Contract No.: N00024-69-C-1268

Title: Kalman Filtering Applied to Track Detection

In any track detection function, the problems of detection and estimation must be addressed. The best estimate of a potential track's future position and velocity are needed for association of the many target-like returns with their individual tracks as they are observed. This technique requires placing a gate (search area) around the predicted position of a particular target, and if a signal return falls within the gate, it is associated with that target. The optimum size and shape of the gates require a solution to the so called association problem, we have used the gain matrix from the Kalman filter along with a velocity sensitive component to set the gate size. The gate size versus ping number with and without this velocity component appears in Figure 1.

The predicted position and velocity is needed for both the positioning of the gate and the detection of the target. In this technical memo the optimal processing of the sonar returns to extract range and velocity information will be reviewed using the approach first described by Kalman (Refs. 1, 2).

The Kalman filter will be applied to a linear system. A linear system is one that obeys the principle of superposition. Physically, superposition implies that the presence of one excitation does not affect the responses due to other excitations; there are no interactions among responses of different excitations within a linear system. In the analysis of the combined returns of different excitations, one can begin by considering each individual cause and effect separately as though all other causes were absent and then sum over all the individual excitations and corresponding effects. Linearity as applied to the random variables $y, \dot{y}$ (measured range and rate rate) means that as independent measurements they will define the total measured state of the system. With this in mind the set of linear differential equations describing the physical system can be written as

$$\frac{dx_i}{dt} = f(x_j, u) \qquad (1)$$

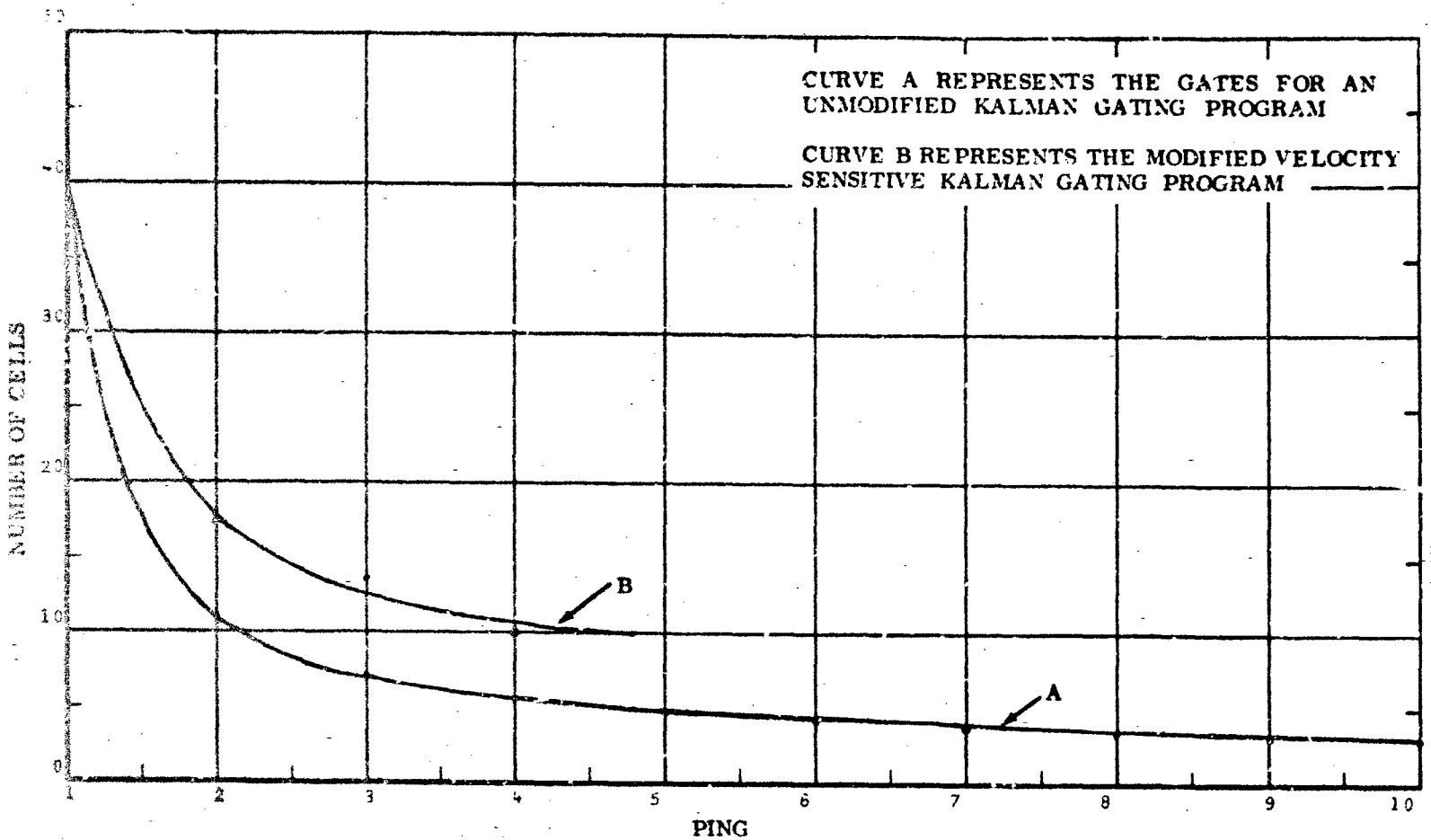

Figure 1. Gate Sizes Used in Correlating Ping-to-Ping Returns

where x_i is the set of random vectors that define the states of the system and u is related to some random noise generator or forcing function that adds a noise component to the measured states. The measured state vector is then expressed as,

$$\underline{y_i} = \underline{Hx_i} + \underline{v}$$

where $\underline{v}$ is the gaussian zero mean noise vector.

Equation (1) can be written in matrix notation as follows.

$$\dot{\underline{x}}(t) = F(t)\, x(t) + D(t)\, U(t)$$

with solution to the discrete-time system (see Appendix B) given by

$$\underline{x}(t+1) = \Phi(t+1,\ t)\ \underline{x}(t) + \Delta(t+1,\ t)\ \underline{U}(t) \qquad (2)$$

Depending on the number of state variables used to describe the linear dynamic system, Equations (1) and (2) take on different forms.

The two-state case is represented by the following set of linear differential equations.

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = u_x$$

In matrix representation, these equations become

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ u_x \end{pmatrix}$$

where x_1 is the position and x_2 the range-rate of a target. Solving these equations for the state vector, we find that the transition matrix Φ and distribution matrix Δ become, respectively (see Appendix A),

$$\Phi\ (t+1,\ t) = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}, \Delta\ (t+1,\ t) = \begin{pmatrix} 0 & \frac{t^2}{2} \\ 0 & t \end{pmatrix}$$

and

$$\begin{pmatrix} x_1(t+1) \\ x_2(t+1) \end{pmatrix} = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 0 & \frac{t^2}{2} \\ 0 & t \end{pmatrix} \begin{pmatrix} 0 \\ u_x \end{pmatrix}$$

The three-state case is represented by

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = x_3$$

$$x_3 = u_x$$

In matrix representation these equations become

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ u_x \end{pmatrix}$$

where x_3 is the target's acceleration. For the three-state case, the state vector can be shown to have the following form.

$$\begin{pmatrix} x_1(t+1) \\ x_2(t+1) \\ x_3(t+1) \end{pmatrix} = \begin{pmatrix} 1 & t & \frac{t^2}{2} \\ 0 & 1 & t \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} + \begin{pmatrix} 0 & 0 & \frac{t^3}{2} \\ 0 & 0 & \frac{t^2}{2} \\ 0 & 0 & t \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ u_x \end{pmatrix}$$

Having described the equations of motion for the linear dynamic system, some properties of the random variables that are required by the filter equation will now be given.

As mentioned above, while defining the random variables in Equation (2), u(t) is a gaussian random vector. This means that the sequence of random vectors x(t - 1), x(t + 1), generated by Equation (2), is a Gauss-Markov sequence. Therefore, the best estimate of a future state can be made without knowledge of all previous history. Since the development of the equations in the Kalman filter depended on this property, it is not surprising that this filter gives a best estimate of a future state based on the last observation of the states. To complete the picture of the model for the system, since u(t) is gaussian, the sequence of random vectors u(t - 1), u(t), and u(t + 1) are normally distributed such that the cross-covariance matrix,

$$\text{cov}\left[u(t_1),\ u(t_2)\right] = 0$$

This implies that $u(t_1)$ and $u(t_2)$ are independent.

Furthermore, it is assumed that $E[u(t)] = 0$. Since this noise generator is present, the output vector $y(t)$ of the measured signal alone must be modified by some additive white noise (gaussian), v, with $E[v(t)] = 0$, $E[v(t_1) \cdot v(t_2)] = 0$, and $E[v(t) \cdot v(t)T] = \text{cov}[v(t_1)] = R(t)$. Physically, this means that $v(t_1)$ and $v(t_2)$ are independent and that a considerable amount of inaccuracy is present in the measurement of the signal. In fact, the further one looks back into the state's history, the larger the indeterminacy.

The filtering problem then is defined as giving the best estimate of all states of the system based on the noisy observation of the observable states. Since the system is linear, the estimated states are expressed as:

$$\underline{\hat{x}}_n = \underline{\tilde{x}}_n + K(y_n - H\underline{\tilde{x}}_n) \tag{3}$$

where $K(t)$ is a continuously updated weighting factor imposed on the variance between the observed and predicted states and as yet has not been defined explicitly. $\hat{x}_n(t)$ is the best estimate of $x_n(t)$ based on the current observation $y_n(t)$.

In order to determine $K(t)$ it is necessary to minimize the mean squared estimation error

$$L = E[(x - \hat{x}) \cdot (x - \hat{x})^T] \tag{4}$$

Since the errors are gaussian, minimizing Equation (4) will produce an optimal weighting (gain) matrix. It is in this sense that the Kalman filter is optimal; however, this property alone will not necessarily achieve considerably higher levels of performance than some less "optimal" filtering method. By performance other authors usually refer to the rate at which the elements of the covariance matrix decrease. In other words, with increasing time the estimates of the system are said to improve. For the case when there are several data sample sets however, the updated estimate of the gate sizes ("updated" here means that the size of the gates is decreasing) can in effect be reducing the probability of declaring a false track, by eliminating certain nonconsistent tracks. At the same time, if the target motion is not consistent with the system equations of motion, the detection probability of the system will be decreased.

Using Equation (3), the loss function becomes

$$E[(x - \hat{x}) \cdot (x - \hat{x})^T] = E[(x - \tilde{x}) \cdot (x - \tilde{x})^T] + E[(K(y - H\tilde{x}))(K(y - H\tilde{x}))^T]$$
$$- E[K(y - H\tilde{x})(x - \tilde{x})^T] - E[(x - \tilde{x}) K^T (y - H\tilde{x})^T]$$

Using the following equations, $y = H\underline{x} + \underline{v}$, and noting that $E[(x - \hat{x})\, v^T] = 0$, the loss function now becomes

$$E[(x - \hat{x})(x - \hat{x})^T] = P - KHP - PH^T K^T + K(HPH^T + R)\, K^T$$

In order to minimize the loss function we set the derivative of L, w · r · t · K, dL/dK, equal to zero. In so doing,

$$-2PH^T + 2(HPH^T + R)\, K = 0$$

Solving this equation for the gain matrix,

$$K = PH^T (HPH^T + R)^{-1}$$

With this value for the gain matrix the estimated covariance matrix becomes

$$\hat{P} = (I - KH)P$$

In order to complete the filtering problem a recursive relationship for the conditional covariance matrix P(t + 1, t) must be derived.

$$P(t + 1, t) = E[(x(t + 1) - \tilde{x}(t + 1, t)), (x(t + 1) - \tilde{x}(t + 1, t))^T] \tag{6}$$

but

$$x(t + 1) = \Phi(t + 1, t)\, x(t) + \Delta(t + 1, t)\, u(t) \tag{7}$$

and

$$\tilde{x}(t + 1, t) = \Phi(t + 1, t)\, \hat{x}(t + 1, t) \tag{8}$$

After substituting Equations (7) and (8) into Equation (6), the predicted covariance matrix becomes

$$P(t + 1) = \Phi(t + 1, t)\, \hat{P}(t)\, \Phi(t + 1, t)^T + \Delta E[u \cdot u^T]\, \Delta^T$$

For the two-state case where the model error is assumed constant, the predicted covariance matrix becomes

$$\tilde{P}(t + 1) = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \sigma^2_{r_{init}} & 0 \\ 0 & \sigma^2_{v_{init}} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ t & 1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ \sigma^2_v \end{pmatrix}$$

$$= \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix}$$

where

$$P_{11} = \sigma_{r_{init}}^2 + \sigma_{v_{init}}^2 t^2$$

$$P_{12} = P_{21} = \sigma_{v_{init}}^2 t$$

$$P_{22} = \sigma_{v_{init}}^2 + \sigma_v^2$$

$$K = \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \left[(1 \quad 0) \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \sigma_{r_{init}}^2 \right]^{-1}$$

$$= \begin{pmatrix} P_{11} \\ P_{21} \end{pmatrix} \left(P_{11} + \sigma_{r_{init}}^2 \right)^{-1}$$

$$\hat{P} = \begin{pmatrix} 1 - \dfrac{P_{11}}{P_{11} + \sigma_{r_{init}}^2} & 0 \\ - \dfrac{P_{21}}{P_{11} + \sigma_{r_{init}}^2} & 1 \end{pmatrix} \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix}$$

$$= \begin{pmatrix} \dfrac{P_{11}\,\sigma_{r_{init}}^2}{P_{11} + \sigma_{r_{init}}^2} & P_{12} - \dfrac{P_{12}\,P_{11}}{P_{11} + \sigma_{r_{init}}^2} \\ P_{12} - \dfrac{P_{21}\,P_{11}}{P_{11} + \sigma_{r_{init}}^2} & P_{22} - \dfrac{P_{21}\,P_{12}}{P_{11} + \sigma_{r_{init}}^2} \end{pmatrix}$$

With these values, the following can be defined.

$$\sigma_r(t) = P_{11}$$

$$\tilde{\sigma}_v(t) = P_{22}$$

$$\tilde{\sigma}_{r\text{-}\dot{r}} = P_{12} = P_{21}$$

The gate sizes (see Appendix B) for this model appear in Figure 1. Curve A contains no model error, i.e., $\sigma_v^2 = 0$. Curve B contains the model error contribution, σ_v^2, but its value is obtained from the predicted range rate.

For the two-state case with nonconstant model errors, the predicted covariance matrix has the form (set $\sigma_v^2 = 0$).

$$\tilde{P}(t+1) = \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix} + (x_2)^2 \begin{pmatrix} \frac{T^2}{4} & \frac{T}{2} \\ \frac{T}{2} & 1 \end{pmatrix}$$

where x_2 is the expected range rate.

Equation (2), as it stands, does not give the predicted estimate of the system. Since on the average, expected values will result the predicted estimate of the state vector is given by

$$\tilde{X}(t+1) = \Phi(t+1, t)\,\hat{x}(t) + \Delta(t+1, t)\,E[u(t)]$$

However, since $E[u(t)] = 0$ for all t, then

$$\tilde{X}(x+1) = \Phi(t+1, t)\,\hat{x}(t)$$

A summary of the filter equations is given in Figure 2.

In order to start the filtering problem some initialization of the covariance matrix, $P(t_0)$, for the best estimates of the initial states of the model must be made, i.e., some systematic selection of the initial values of the covariance matrix. The selection of these elements depends upon a knowledge of the problem, i.e., the track dynamics. Unfortunately, a hostile target does not cooperate in revealing its maneuvers. In this instance, it seems that the best selection would be to set the off diagonal elements to zero. This is reasonable because knowledge of the initial range (on either the first ping or first return) will in no way provide information about the initial velocity, i.e., initial range and velocity are uncorrelated. The main diagonal terms are set so that the predicted covariance will set a gate about a 6-knot target.

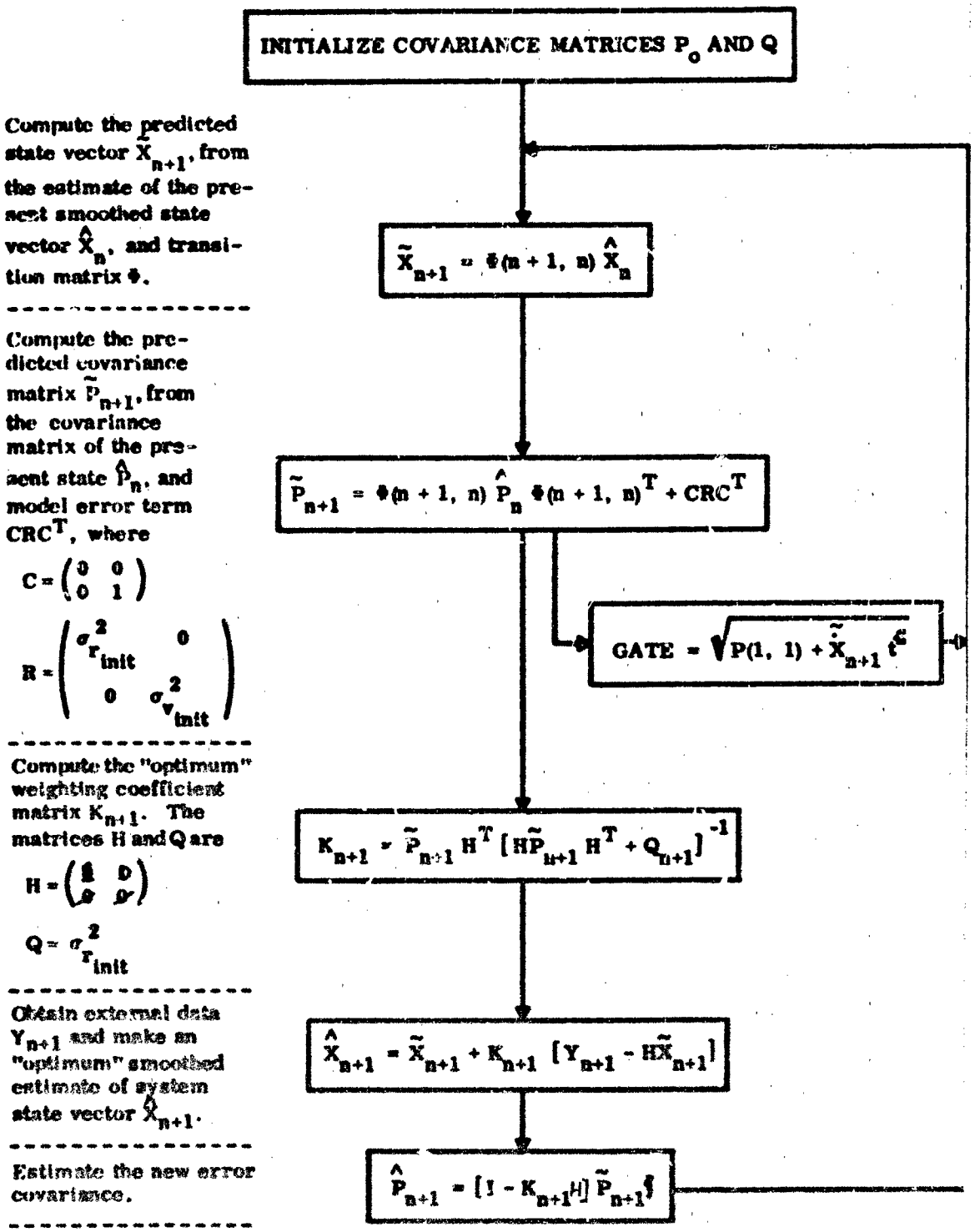

Figure 2. Range Prediction, Track Estimation Kalman Filtering, and Gate Determination

It should be noted that the covariance matrices are predetermined, i.e., their values are independent of any measurements. Clearly this means that the gates are updated (narrowed) for both false and fading targets. In order to rectify this insensitivity to the real world, as a preliminary improvement to the covariance matrix, the calculated predicted range-rate is used in the matrix. Qualitatively, this seems to have reduced the root-mean squared error between the predicted and measured state vector more quickly than when no velocity terms were used in the model errors. More work on this point is presently in progress.

In conclusion, one of the main functions of the Kalman filter is to update the measured state vectors obtained from previous instants of time to a set of values corresponding to the present instant. These updated measurements are used along with the current measurements to form an optimum estimate of the state vector. Since a random disturbance term exists in the canonical equations of motion there is an uncertainty in updating the measurements of the previous instants. The Kalman filter takes this fact into account by assigning weighting factors to the contributions of the previous instants. The fact that the mean squared estimation error is minimized and that the Kalman filter uses matrix weighting factors so that each component in the state vector can be weighted individually, accounts in part for its superiority as a filter.

APPENDIX A

DERIVATION OF THE TRANSITION AND DISTRIBUTION MATRICES FOR THE TWO- AND THREE-STATE CASES

The two-state case:

$$\dot{x} = x_2 \tag{9}$$

$$\dot{x}_2 = u_x \tag{10}$$

Integrating Equation (10) first w · r · t · "t"

$$x_2(t + 1) = u_x T + C_1 \tag{11}$$

and Equation (11) becomes

$$x_2(t + 1) = u_x T + x_2(t) \tag{12}$$

Substituting Equation (12) into Equation (9) and integrating over t, keeping $x_2(t)$ constant,

$$x_1(t + 1) = \frac{u_x T^2}{2} + x_2(t)\, T + C_2 \tag{13}$$

Again, applying the boundary conditions,

$$x_1(t + 1) = \frac{u_x T}{2} + x_2(t)\, T + x_1(t) \tag{14}$$

where $x_1(t)$ is some initial position at T = 0. Therefore, the canonical equations of motion become

$$\begin{aligned} x_1(t + 1) &= x + T_{x_2} + \frac{u_x T^2}{2} \\ x_2(t + 1) &= 0 + x_2 + u_x T \end{aligned} \tag{15}$$

In order to put these equations into matrix notation, first examine the case where $u_x = 0$, i.e., there are no moded errors.

$$\begin{aligned} x_1(t + 1) &= x_1 + T_{x_2} \\ x_2(t + 1) &= 0 + x_2 \end{aligned} \tag{16}$$

Clearly, these equations in matrix notation become

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_t = \Phi(t+1,\ t)\ \hat{x}(t) \tag{17}$$

From the comparison of Equation (15) with Equation (16), it can be seen that another matrix must be added to account for the model errors. These can be represented by the following matrices.

$$\text{Model Contribution} = \begin{pmatrix} 0 & \frac{T^2}{2} \\ 0 & T \end{pmatrix} \begin{pmatrix} 0 \\ u_x \end{pmatrix} = \Delta(t+1,\ t)\ u(t) \tag{18}$$

The derivation of the equations in the three-state case follows the exact same procedure.

APPENDIX B

As a preliminary note on the application of the continuous-time Kalman filter to the discrete-time model, it seems that if the gates close down to their asymptotic value after one ping, then either large model errors have to be included or a nonlinear approach to the filtering problem adopted. The application of the filter to the nonlinear problem is forthcoming.

REFERENCES

1. Kalman, R.E., "A New Approach to Linear Filtering and Prediction Problems," J. Basic Eng., No. 82, pp. 35-45, March 1960.

2. Kalman, R.E., and Bucy, R.S., "New Results in Linear Filtering and Prediction Theory," J. Basic Eng., No. 83, pp. 95-108, March 1961.

3. Jardine, F.D., "Optimal Filter Design for Sampled Data Systems with Illustrative Examples," Thesis, U.S. Naval Postgraduate School, AD-485-275, 1965.

4. Richanran, J., and Thau, F., "Elements of Kalman Filtering Techniques," TNB-General Precision, Inc., No. 9, pp. 17-27, 1966.

5. Sposoto, L.E., "Finite Memory — Recursive and Adaptive Recursive Filters," Syracuse University Research Corp., AD-841-859.