# V4.17

## Computationally efficient tracking of multiple targets by probabilistic data association using neural networks*

DEBASIS SENGUPTA AND RONALD A. ILTIS

Department of Electrical and Computer Engineering
University of California, Santa Barbara
Santa Barbara, CA 93106

**Abstract**

The joint probabilistic data association (JPDA) algorithm has been previously reported to be suitable for the problem of tracking multiple targets in the presence of clutter. Although it make few assumptions and can handle many targets, the complexity of this algorithm increases rapidly with the number of targets and returns. An approximation of the JPDA has been suggested in this paper. The proposed algorithm uses an analog computational network to solve the data association problem. The problem is viewed as that of optimizing a suitably chosen objective function. Simple neural network structures for the approximate minimization of such functions have been proposed by other researchers. The analog network used here offers a significant degree of parallelism and thus can compute the association probabilities more rapidly. Computer simulations indicate the ability of the algorithm to track many targets simultaneously in the presence of moderate density clutter.

### 1. Introduction

The problem of tracking multiple targets in the presence of clutter has derived considerable attention in recent years. The probabilistic data association (PDA) approach reduces the complexity of more sophisticated algorithms by focusing on the few most likely hypotheses [Bar-Shalom 1978]. This is done by forming a validation gate around the predicted position of each target and selecting only the returns inside the gate for association with the target. The algorithm assigns a probability, called the *association probability*, to every hypothesis associating a return to a target. A weighted average of the state estimates under all the hypotheses associating different returns to a particular target serves as the PDA estimate of the state of that target. The association probabilities are used as weights. Assuming the true return from a given target to be a Gaussian random vector with mean and covariance given by the predicted state vector and covariance matrix, the likelihood function can be evaluated under every hypothesis associating different returns to that target. The association probabilities are computed from these likelihoods. The simplest of the PDA methods computes the association probabilities taking one target at a time, regardless of how close the other targets may be. For this reason the ordinary PDA tracker performs poorly while tracking crossing targets or when the targets are close to each other. This difficulty is alleviated by the joint probabilistic data association (JPDA) algorithm which computes the association probabilities from the *joint* likelihood functions corresponding to the joint hypotheses associating *all* the returns to different permutations of targets and clutter points [Fortmann *et al* 1983]. However the increase of complexity in the computation of the association probabilities may prove significant for a number of targets in moderate density clutter. There have been efforts to approach the performance of the JPDA by imitating its properties via *ad hoc* association rules [Fitzgerald 1986]. However the effectiveness of these approximations in tracking many targets in the presence of clutter is not guaranteed.

This paper suggests the use of an analog computational network for computing the association probabilities. The task of finding these probabilities is viewed as a constrained optimization problem. The constraints are obtained by a careful evaluation of the properties of the JPDA association rule. Some of these constraints are analogous to those of the classical "Traveling salesman problem" (TSP) and hence the neural network solution of this problem suggested by Hopfield and Tank [1985] is used as a reference.

### 2. Problem description and the PDA solution

$T$ dynamic systems of the following familiar form are considered.

$$x^t(k+1) = F^t(k)x^t(k) + w^t(k), \qquad t \leq T \tag{1}$$

where $x$ is the plant or target state vector, $F$ is a known matrix describing the dynamics of the target and $w$ is a vector of zero-mean Gaussian noise uncorrelated to any such noise vector at a different instant of time. The superscript $t$ corresponds to the $t$th target and $k$ is the index of time. The plant noise vectors $w^t(k)$ for different targets are assumed to be uncorrelated with each other at all instants of time $k$. The measurement model is

$$z(k) = H(k)x^t(k) + v(k) \tag{2}$$

where $z$ is the measurement vector, $H$ is a known matrix and $v$ is a zero-mean Gaussian noise vector independent of $w^t$. The vectors $v$ at different time instants are assumed to be independent of each other. The covariance matrices $Q^t(k)$ and $R(k)$ of $w^t(k)$ and $v(k)$, respectively, are known. The initial state of the target $t$, $t = 1, 2, \cdots, T$ is assumed to be Gaussian with known mean vector $\hat{x}^t(0|0)$ and known covariance matrix $P^t(0|0)$.

The PDA approach considers only a subset of measurements which are validated by an appropriate gate. The $l$th return $z_l(k)$ is validated for the $t$th target if its Mahalanobis distance from the predicted location $\hat{x}^t(k|k-1)$ of the $t$th target is less than a threshold, *i.e.*,

$$r_l^t(k)^2 = [z_l(k) - H(k)\hat{x}^t(k|k-1)]' W^t(k)^{-1} \cdot [z_l(k) - H(k)\hat{x}^t(k|k-1)] \leq \gamma^2 \tag{3}$$

where

$$W^t(k) = H(k)P^t(k|k-1)H(k)' + R(k) \tag{4}$$

* This work was sponsored in part by the SDIO/IST program managed by the Office of Naval Research under contract no. N00014-85-K-0551.

$P^t(k|k-1)$ is the predicted error covariance for the $t$th target and the threshold $\gamma$ is selected to produce a predetermined probability of erroneous rejection of the correct return. It is assumed for simplicity that the true return is a Gaussian random vector with mean $H(k)\hat{x}^t(k|k-1)$ and covariance $P^t(k|k-1)$. A validation matrix may be formed by using (3) in the following way.

$$[\Omega(k)]_{l,t} = \begin{cases} 1 & \text{if the } l\text{th return belongs to } G^t(k) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where $G^t(k)$ is the validation region of the $t$th target. $\Omega(k)$ has dimension $m(k) \times T$ where $m(k)$ is the total number of validated returns at the $k$th instant and $l$ indexes the set of these returns.

The PDA state estimate is given by a weighted combination of all the estimates under different hypotheses.

$$\hat{x}^t(k|k) = \sum_{l=0}^{m(k)} \beta_l^t(k)\hat{x}_l^t(k|k) \quad (6)$$

where $\hat{x}_l^t(k|k)$ is the state estimate under the hypothesis that the $l$th validated return came from the $t$th target. The weight $\beta_l^t(k)$ corresponds to the probability that the $l$th return came from the $t$th target. The overall covariance update is

$$P^t(k|k) = \sum_{l=0}^{m(k)} \beta_l^t(k)P_l^t(k|k) \quad (7)$$

where $P_l^t(k|k)$ is the covariance matrix corresponding to $\hat{x}_l^t(k|k)$.

The various PDA techniques differ only in the ways they compute the association probabilities $\beta_l^t(k)$ which must satisfy

$$\sum_{l=0}^{m(k)} \beta_l^t(k) = 1, \quad t = 1, 2, \cdots, T \quad (8)$$

The ordinary PDA focuses only on the returns in one validation region at a time. If the probability of detecting the correct return is $P_D$ and the probability of validating a detected return is $P_G$ then it can be shown that [Fortmann *et al* 1983] the PDA association probabilities are

$$\beta_l^t(k) = \frac{p_l^t(k)}{\sum_{l=0}^{m(k)} p_l^t(k)} \quad (9)$$

where

$$p_l^t(k) = \begin{cases} \lambda(1 - P_D P_G) & \text{if } l = 0 \\ \frac{P_D}{(2\pi)^{M/2}|W^t(k)|^{1/2}} \exp[-\frac{1}{2} r_l^t(k)^2] & \text{if } [\Omega(k)]_{l,t} = 1 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

Computation of the JPDA association probabilities is more complicated. Define a feasible hypothesis of *joint* events $\chi(n,k)$ as a permutation of targets and clutter points which allows at most one target to be associated with a given return and at most one return to be associated with a given target. It was shown by Fortmann *et al* [1983] that the JPDA association probabilities are

$$\beta_l^t(k) = \sum \frac{\lambda^{m(k)-T}}{c} \prod_{\tau=1}^{T} p_{j^\tau}^\tau(k) \quad 1 \le \tau \le T, \quad 0 \le l \le m(k) \quad (11)$$

where $j^\tau$ is the return to which the $\tau$th target is associated under a given hypothesis and the sum is over all feasible hypotheses which allow the $l$th return to be associated with the $t$th target. In the above equation $p_l^t$ is as described in (10) except for a minor difference in the case of $p_0^t$ which is given by $\lambda(1 - P_D)$. $c$ is a normalizing constant. (11) reflects the fact that the JPDA uses the joint likelihoods of *all* the returns while (9) shows that the ordinary PDA effectively uses a normalized version of the simple likelihood functions as the association probabilities.

It is useful to examine the properties of $\beta_l^t(k)$ which are exclusive to the JPDA. It is unlikely that $\beta_l^t(k)$ will be large if $\beta_l^\tau(k)$ for some $\tau \neq t$ is also large. Furthermore, a close examination of (11) reveals that $\beta_l^t(k)$ is large if, in addition to $p_l^t(k)$ being large, there is a highly likely combination of the remaining targets $\{\tau : 1 \le \tau \le T, \tau \neq t\}$ and the remaining returns $\{j : 1 \le j \le m(k), j \neq l\}$. A highly likely combination is indicated by a large value of the product of the corresponding $p_l^t(k)$s. This property of the JPDA association probabilities was not considered in the following *ad hoc* formula for the $\beta_l^t(k)$s proposed by Fitzgerald [1986].

$$\beta_l^t(k) = \frac{p_l^t(k)}{\sum_l p_l^t(k) + \sum_t p_l^t(k) - p_l^t(k) + p_0} \quad (12)$$

The additional term $\sum_t p_l^t(k)$ in the denominator (compare (9)) is introduced in order to reduce the chances of $\beta_l^t(k)$ and $\beta_l^\tau(k)$, $\tau \neq t$ being large at the same time. The arbitrary constant $p_0$ is expected to improve the performance in the presence of clutter. Another PDA scheme, called the *nearest neighbor* PDA (NNPDA), picks the joint hypothesis which best explains the ordinary PDA association probabilities described by (9). The resulting matrix of $\beta(k)$s consists only of 1s and 0s. The task of choosing the nearest hypothesis may require an exhaustive search over all feasible hypotheses. All of the above PDA techniques can track two targets satisfactorily. However their performance in tracking more targets in the presence of moderate density clutter may be inadequate.

The complexity of the JPDA is overwhelming for relatively large $T$ and $m(k)$, and yet a computationally efficient substitute based on a careful understanding of its properties is lacking. The next section develops a method which is expected to serve as an effective alternative to the JPDA tracking scheme.

## 3. The traveling salesman problem and the JPDA

The traveling salesman problem (TSP) is a well-known constrained multidimensional optimization problem. Given a set of $n$ cities and the distance between each pair of them, the task is to design a closed tour for a traveling salesman. The length of the tour should be minimized subject to the constraints that no city should be excluded or visited twice. For an $n$-city TSP, there are $n!/2n$ distinct valid tours out of which the shortest has to be chosen. Hopfield and Tank [1985] proposed a neural network solution to the TSP by using $n^2$ interconnected neurons. They formulated the problem as that of minimizing an objective (or *energy*) function of the form

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} T_{ij}V_iV_j - \sum_{i=1}^{N} V_iI_i \quad (13)$$

over all $V_i$. They also designed an analog circuit which approximately minimizes the above energy function. $T_{ij}$ represents the strength of connection from the output of the $j$th neuron to the

input of the $i$th neuron while $I_i$ represents the externally supplied input current to the $i$th neuron. The evolution of the circuit is described by the differential equation

$$\frac{du_i}{ds} = -\frac{u_i}{s_o} + \sum_{j=1}^{N} T_{ij} V_j + I_i \tag{14}$$

where $s$ is the index of time, $s_o$ is a time constant and $u_i$ is the input to the $i$th neuron. The output $V_i$ of the neuron is related to its input by a monotonically nondecreasing function $g$ with domain $(-\infty, \infty)$ and range $[0,1]$. Although the circuit operates over the $N$-dimensional hypercube defined by $0 \le V_i \le 1$, the minima of (13) occur only at the corners of the hypercube, provided the connections are symmetric ($T_{ij} = T_{ji}$). In the case of TSP a proper choice of a few design parameters causes the network to converge to a stable state or minimum which corresponds to one of the best few solutions.

The problem of finding the association probabilities $\beta_l^t(k)$ from the likelihoods $p_l^t(k)$ described in the previous section has features similar to the TSP. If the output voltages of an $(m(k)+1) \times T$ array of neurons are defined to be the association probabilities, then the columns would represent targets and the rows would indicate returns. The zeroth row would correspond to no return from a given target. The constraints of the data association problem (DAP) are as follows. Each column sum of the voltages must be unity so that (8) is satisfied. At most one large entry may be favored in every row and column in accordance with the assumptions that no two returns can come from the same target and no return can come from two targets. $\beta_l^t(k)$ should be large if $p_l^t(k)$ is large and there is a combination of the remaining targets and returns which result in a large product of the corresponding likelihoods. All these requirements are accomplished by minimizing the energy function

$$\begin{aligned} E_{DAP} = {} & \frac{A}{2} \sum_l \sum_t \sum_{r \neq t} V_l^t V_l^r + \frac{B}{2} \sum_l \sum_t \sum_{j \neq l} V_l^t V_j^t \\ & + \frac{C}{2} \sum_t \left( \sum_l V_l^t - 1 \right)^2 + \frac{D}{2} \sum_l \sum_t \left( V_l^t - \rho_l^t \right)^2 \\ & + \frac{E}{2} \sum_l \sum_t \sum_{r \neq t} \left( V_l^t - \sum_{j \neq l} \rho_j^r \right)^2 \end{aligned} \tag{15}$$

where $\rho_l^t$ is a normalized version of the $p_l^t(k)$s,

$$\rho_l^t = \frac{p_l^t(k)}{\sum_{l=0}^{m(k)} p_l^t(k)}, \quad l = 0, 1, \cdots, m(k) \tag{16}$$

$\rho_l^t$ is roughly the PDA association probability for the $l$th return and $t$th target (compare (9)). The $\beta_l^t(k)$s are obtained as the output $V_l^t$, $l = 0, 1, \cdots, m(k)$, $t = 1, 2, \cdots, T$. The index $k$ is dropped for convenience. The first term achieves its minimum value of 0 when there is at most one large entry in each row. Similarly the second term inhibits more than one return to be strongly associated with a given target. The third term biases the final solution towards a normalized set of numbers. The fourth term favors associations which have a high likelihood and the fifth term favors a highly likely combination of the remaining targets and returns. Minimizing (15) can be shown to be equivalent to minimizing the energy function

$$E'_{DAP} = -\frac{1}{2} \sum_l \sum_j \sum_t \sum_r T_{lj}^{tr} V_l^t V_j^r - \sum_l \sum_t V_l^t I_l^t \tag{17}$$

where

$$\begin{aligned} T_{lj}^{tr} = - \big[ & A\delta_{lj}(1-\delta_{tr}) + B\delta_{tr}(1-\delta_{lj}) + C\delta_{tr} + D\delta_{tr}\delta_{lj} \\ & + E(T-1)\delta_{tr}\delta_{lj} \big] \end{aligned} \tag{18}$$

and

$$I_l^t = C + (D+E)\rho_l^t + E(T - 1 - \sum_r \rho_l^r) \tag{19}$$

Here $\delta_{ij}$ is the usual Kronecker delta function. Equation (17) is equivalent to (13) except that each neuron is identified by a superscript as well as a subscript. It may be observed that the strengths of connection $T_{lj}^{tr}$ given by (18) do not depend on $\rho_l^t$. Thus a new network need not be designed every time a new set of likelihood functions are available. Instead, the input currents $I_l^t$ as described by (19) can be easily controlled by these functions. The equations of motion for this circuit are

$$V_l^t = g(u_l^t) \tag{20a}$$

$$\begin{aligned} \frac{du_l^t}{ds} = {} & -\frac{u_l^t}{s_o} - A \sum_{\substack{r=1 \\ r \neq t}}^{T} V_l^r - B \sum_{\substack{j=0 \\ j \neq l}}^{m(k)} V_j^t - C \left( \sum_{j=0}^{m(k)} V_j^t - 1 \right) \\ & - [D + E(T-1)] V_l^t + (D+E)\rho_l^t \\ & + E \left( T - 1 - \sum_{r=1}^{T} \rho_l^r \right) \end{aligned} \tag{20b}$$

The coefficients $A$, $B$, $C$, $D$ and $E$ can be adjusted to control the emphasis on different constraints and properties. A large value of $D$ will produce $V_l^t$ close to $\rho_l^t$ which are approximately the ordinary PDA association probabilities. A larger emphasis on $A$, $B$ and $C$ will produce the nearest neighbor solution (as in the TSP). This special case offers an attractive method of determining the nearest hypothesis without explicitly examining all the hypotheses. Finally, a balanced combination of all five terms will lead to the most complete emulation of all the properties of the JPDA. A larger number of targets will only require a larger array of interconnected neurons instead of an increased load on any sequential software to compute the association probabilities. Figure 1 shows the block diagram of a tracker which uses the proposed analog network to compute the association probabilities.

## 4. Simulation of performance

The performance of the neural network PDA (NPDA) was simulated along with those of the Fitzgerald's simplified PDA (SPDA) and JPDA. Four targets were chosen with nearly constant velocities in a two dimensional plane. The discretized state equation for each target is given by (1) where

$$x^t(k) = (\, x \quad \dot{x} \quad y \quad \dot{y} \,)' \tag{21}$$

$$F^t(k) = \begin{pmatrix} 1 & \Delta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \forall\, t, \ \forall\, k \tag{22}$$

using Cartesian coordinates in two dimensions. The quantities in the right hand side of (21) indicate the position and velocity at time $k$ and target $t$. The sampling interval ($\Delta$) was assumed to be 0.5 second. The plant noise $w^t(k)$ is assumed to be of the form [Fortmann 1983]

$$w^t(k) = \begin{pmatrix} \Delta/2 & 0 \\ 1 & 0 \\ 0 & \Delta/2 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} w_x^t(k) \\ w_y^t(k) \end{pmatrix}$$

where $(w_x^t(k)\ w_y^t(k))'$ is a vector of independent Gaussian velocity noises with associated variances $(\sigma_x^t(k))^2 = (\sigma_y^t(k))^2 = 0.0009\ \mathrm{km^2 s^{-2}}\ \ \forall\, t,\ \forall\, k$ which correspond to a standard deviation of at least .04 times the mean velocity. These uncertainties allow for small changes of the target velocity. It was assumed that only position measurements are available so that

$$H(k) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

for all $k$. The measurement noise covariance matrix is

$$R(k) = diag(.0225, .09)$$

assuming all the measurement noise to be uncorrelated. The probability of validation $P_G$ and the probability of detection $P_D$ were both assumed to be 0.95. The corresponding threshold of the Mahalanobis distance, as obtained from the table of $\chi_2^2$ distribution is $\gamma = \sqrt{9.2}$. This determines the size of the validation gate. A uniform clutter density of about 0.05 $\mathrm{km^{-2}}$ was chosen. This produced on the average 0.35 clutter point per validation gate.

Equations (20*a*) and (20*b*) represent the evolution of the states of the analog circuit for the neural network solution of the DAP. The function $g(u_l^t)$ in (20*a*) was defined to be [Hopfield and Tank 1985]

$$V_l^t = g(u_l^t) = \frac{1}{2}\left(1 + \tanh\frac{u_l^t}{u_o}\right) \tag{23}$$

In order to avoid bias to any particular set of stable states, the initial states were chosen to be such that they approximately produce $\sum_{l=0}^{m(k)} V_l^t = 1\ \forall\ t$ initially. It is known [Hopfield and Tank 1985] that the addition of a random noise to the initial values enhances convergence. Accordingly the initial states were chosen to be

$$u_l^t = \delta u_l^t - \frac{u_o}{2}\ln m(k), \quad l \le m(k),\ t \le T$$

where $\delta u_l^t$ is a random variable having uniform distribution over $[-0.1u_o,\ 0.1u_o]$. The constant $u_o$ was chosen to be 0.02 to produce a moderate rate of convergence. The differential equation (20*b*) was approximated by a difference equation with stepsize 0.0005 s. $s_o$ was selected to be 1 s. An experimentally determined limit of 100 iterations was imposed on the recursive difference equation. The 100th iterates $V_l^t(100)$, $l \le m(k)$, $t \le T$ are expected to be close to the steady state solutions of (20*b*). These values were normalized for each $t$ to ensure strict compliance with the constraint (8). The constants $A$, $B$, $C$, $D$ and $E$ were also chosen experimentally to produce a stable operating point. The values $A = 1$, $B = 40$, $C = 1000$, $D = 30$ and $E = 5$ appeared to be suitable for two to four targets.

Figure 2 shows the ability of the NPDA and the failure of the SPDA to track four targets in the presence of moderate density clutter. The SPDA association probabilities were computed from (12) with $p_0 = 0$. No non-zero value of $p_0$ appeared to improve its performance. The performance of the NPDA is equivalent to that of the JPDA in terms of rms position and velocity errors [Sengupta and Iltis 1987].

## 5. Conclusion

Computation of the association probabilities by an analog network appears to be a viable alternative to the JPDA. It outperforms the SPDA by accurately emulating all the properties of the JPDA. The coefficients of the function $E_{DAP}$ have to be chosen suitably for a given number of targets and returns. Computer simulations indicate that a suitable operating point may be obtained for a range of complexity of the problem. The analog network is structurally simple and there is a high degree of parallelism inherent in it. A DAP of higher complexity will only require a larger array of neurons instead of increased burden on a sequential algorithm such as the JPDA. In spite of the necessity of D/A and A/D conversions, these attractive features may render the NPDA useful for tracking a reasonable number of targets in the presence of moderate density clutter.

## References


1. Y. Bar-Shalom, "Tracking methods in a multitarget environment," Survey paper, IEEE Transactions on Automatic Control, Vol. AC-23, No. 4, Aug. 1978, pp. 618–626.
2. R.J. Fitzgerald, "Development of practical PDA logic for multitarget tracking by microprocessor," Proc. American Controls Conference, Seattle, WA, 1986, pp. 889–898.
3. T.E. Fortmann, Y. Bar-Shalom and M. Scheffe, "Multitarget tracking using joint probabilistic data association," IEEE Journal of Oceanic Eng., July 1983.
4. J.J. Hopfield and D.W. Tank, ""Neural" Computation of decisions in optimizing problems," Biological Cybernetics, Vol. 52, 1985, pp. 141–152.
5. D. Sengupta and R.A. Iltis, "Computationally efficient tracking of multiple targets by probabilistic data association using neural networks," Submitted to IEEE Trans. on Aerospace and Electronic Systems, 1987.


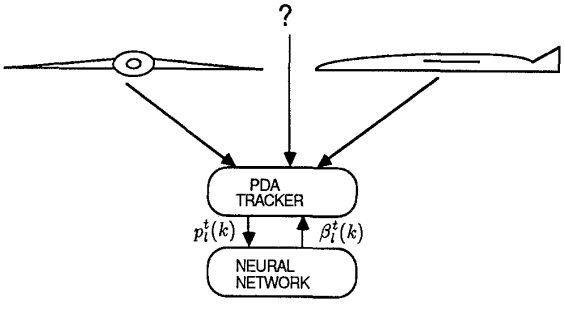


Figure 1: Schematic diagram of NPDA tracker

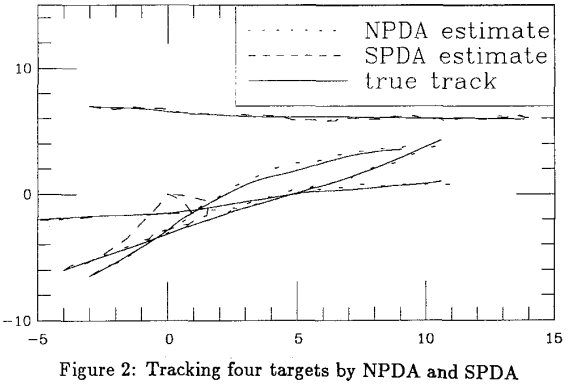


Figure 2: Tracking four targets by NPDA and SPDA