01 Jan 1994

Use of Hopfield Neural Networks in Optimal Guidance

S. N. Balakrishnan
Missouri University of Science and Technology, bala@mst.edu

James Edward Steck

Recommended Citation

S. N. Balakrishnan and J. E. Steck, "Use of Hopfield Neural Networks in Optimal Guidance," *IEEE Transactions on Aerospace and Electronic Systems*, Institute of Electrical and Electronics Engineers (IEEE), Jan 1994.
The definitive version is available at https://doi.org/10.1109/7.250431

[5] Moose, R. L. (1975)
An adaptive state estimation solution to the maneuvering target problem.
IEEE Transactions on Automatic Control, **AC-20** (June 1975), 359–362.

[6] Gholson, N. H., and Moose, R. L. (1977)
Maneuvering target tracking using adaptive state estimation.
IEEE Transactions on Aerospace and Electronic Systems, **AES 13** (May 1977), 310–317.

[7] Ricker, G. C., and Williams, J. R. (1978)
Adaptive tracking filter for maneuvering targets.
IEEE Transactions on Aerospace and Electronic Systems, **AES-14** (Jan. 1978), 185–193.

[8] Thomas, H. W. (1979)
Maneuver handling in multiradar ATC system.
In *Proceedings of the IEE*, **126** (June 1979), 469–475.

[9] Moose, R. L., Vanlandingham, H. F., and McCabe, D. H. (1979)
Modeling and estimation for tracking maneuvering targets.
IEEE Transactions on Aerospace and Electronic Systems, **AES-15** (May 1979), 448–456.

[10] Chan, Y. T., Hu, A. G. C., and Plant, J. B. (1979)
Kalman filter based tracking scheme with input estimation.
IEEE Transactions on Aerospace and Electronic Systems, **AES-15** (Mar. 1979), 237–244.

[11] Bar-Shalom, Y. (1982)
Variable dimension filter for maneuvering target tracking.
IEEE Transactions on Aerospace and Electronic Systems, **AES-18** (Sept. 1982), 621–629.

[12] Chang, C. B., and Tabaczynski, J. A. (1984)
Application of state estimation to target tracking.
IEEE Transactions on Automatic Control, **AC-29**, 2 (Feb. 1984), 98–109.

[13] Blackman, S. S. (1986)
Multiple-Target Tracking With Radar Applications.
Dedham, MA: Artech House, 1986.

[14] Bar-Shalom, Y., and Fortmann, T. E. (1988)
Tracking and Data Association.
New York: Academic Press, 1988.

[15] Bar-Shalom, Y. (1990)
Multitarget-Multisensor Tracking: Advanced Applications.
Boston: Artech House, 1990.

[16] Bogler, P. L. (1990)
Radar Principles with Applications to Tracking Systems.
New York: Wiley, 1990.

[17] Kuczewski, R. (1987)
Neural network approaches to multi-target tracking.
In *Proceedings of the IEEE ICNN Conference*, 1987.

[18] Sengupta, D., and Iltis, R. A. (1989)
Neural solution to the multitarget tracking data association problem.
IEEE Transactions on Aerospace and Electronic Systems, **AES-25** (Jan. 1989), 96–108.

[19] Chin, L. (1992)
Application of neural networks in data fusion.
In *Proceedings of the IEEE International Conference on Intelligent Control and Instrumentation*, Singapore, Feb. 1992.

[20] Hopfield, J. J., and Tank, D. W. (1985)
Neural computation of decisions in optimization problems.
Biological Cybermatics, **52** (1985), 141–152.

[21] Hopfield, J. J. (1982)
Neural networks and physical systems with emergent collective computational abilities.
In *Proceedings of the National Academy of Science*, **79**, 1982, 2554–2558.

[22] Mital, D. P., and Chin, L. (1993)
Intelligent control application with neural networks.
In M. M. Gupta and N. K. Sinha (Eds.), *IEEE Press Book on Intelligent Control Systems: Theory and Practice.*
New York: IEEE Press, Nov. 1993.

[23] Kosko, B. (1992)
Neural Network and Fuzzy Systems.
Englewood Cliffs, NJ: Prentice-Hall, 1992.

Use of Hopfield Neural Networks in Optimal Guidance

A Hopfield neural network architecture is developed to solve the optimal control problem for homing missile guidance. A linear quadratic optimal control problem is formulated in the form of an efficient parallel computing device known as a Hopfield neural network. Convergence of the Hopfield network is analyzed from a theoretical perspective, showing that the network, as a dynamical system, approaches a unique fixed point which is the solution to the optimal control problem at any instant during the missile pursuit. Several target-intercept scenarios are provided to demonstrate the use of the recurrent feedback neural net formulation.

I. INTRODUCTION

The basic problem addressed here is that of guiding a missile to pursue and intercept a moving and most likely accelerating target. Major difficulties in achieving better performance with a modern control based homing missile guidance have been due to the problem of estimating the unknown target acceleration and the nonlinearity of the dynamics and the process of target measurement.

The most popular homing missile guidance is based on a control law called proportional navigation [1]. The basic notion is that if the line-of-sight rate is nulled, then the missile (for a nonmaneuvering, constant velocity target) is on a collision course. If the target is considered smart or able to maneuver perpendicular to the line-of-sight, then variations to the proportional navigation are required for improved performance. These variations have been given theoretical basis through formulations as linear quadratic Gaussian (LQG) problems [2, 3].

Manuscript received January 21, 1993; revised May 3, 1993.

IEEE Log No. T-AES/30/1/13057.

0018-9251/94/$4.00

The LQG formulation allows the design of the controller and the estimator to be separable. However, such separation is completely valid only in the case where the measurement functions are also linear. In a target-intercept problem, the typical measurements such as range, range rate, and bearing angles are all nonlinear functions in a rectangular Cartesian state space. Consequently, the filter structure is suboptimal and the separability assumption of the controller and the estimator is also approximate. Active research has been in progress in this area for improved results under such nonlinear conditions [4–6]. An excellent survey of the available methods that deal with different aspects of the air-to-air missile guidance technology can be found in [7]. As compared with the notions supporting the methods in the available literature, a new approach has been adopted in this study.

The motivation for the method comes from the field of artificial neural networks [8–14]. Artificial neural networks are trainable, highly parallel computational devices which have been successfully used in difficult artificial intelligence tasks such as pattern-recognition, image processing, and optimization. Development of neural networks for adaptive control, estimation and the supporting theory are still emerging [15]. A neural network structure for the uncertain target-intercept dynamics is developed in Section II. This involves posing the problem as a linear quadratic optimal control problem. The feasibility of using a neural network to solve this optimal control problem is shown in the following sections.

II. ARTIFICIAL NEURAL NETWORK STRUCTURE FOR CONTROLLER-ESTIMATOR

A. Artificial Neural Network

Artificial neural networks are modeled after the way the human brain is perceived to function [8]. They consist of a multilayered network of parallel processing elements which are interconnected in such a way that they can modify their behavior in response to their environment. In many neural network applications a neural network is trained with sets of inputs so as to produce a set of known desired outputs (learning phase). If the networks are properly designed and trained, they are insensitive to small variations in its input functions. Consequently, the noise and the distortions of the input will not affect the neural network from producing the proper patterns (response phase).

The application considered in this study, however, is different. The target-intercept problem consists of control of an unknown and unpredictable dynamical system. In this study, an artificial neural network is designed to perform the function of an optimal controller. It does not "learn" specific information but the massively parallel processing capability of the network allows it to rapidly adapt to the unpredictable behavior of the target.

Certain types of artificial neural networks are structured to function primarily in one of two modes. In the first mode, such as a feed-forward network trained with back-propagation [14], all of the processing and information transfer occurs in a forward direction from a group of input neurons to a group of output neurons, and training occurs by providing the network with correct answers at the output. In the second approach such as Hopfield nets [12], there can be feedback paths from any neuron directly to any other neuron in the network. For such a network, considered as a stable dynamical system, the forward and feedback paths will cause the processing of the network to converge to a stable fixed point. It is the feature of feedback that makes the Hopfield nets a feasible approach to solve problems in the area of optimization. The network structure is attractive because it uses a massively parallel computing architecture.

The Hopfield network used here is composed of n neurons where the jth neuron has a calculated net input value u_j^t and output activation value x_j^t at time t given by the discrete time difference equations

$$\begin{aligned} u_i^{t+\Delta t} &= \sum_{j=1}^{n} W_{ij} x_j^t + V_i \\ x_i^{t+\Delta t} &= f(u_i^{t+\Delta t}) \end{aligned} \tag{1}$$

where W_{ij} is the strength or weight of the connection from neuron j to neuron i, and V_i is an external input to the neuron, and the function f is a nonlinear (usually monotonically increasing and bounded) activation function. Each neuron essentially sums weighted inputs from other neurons to calculate its new internal activation u_i, applies an activation function to this value, and broadcasts this value along the connections to other neurons. If the interconnection weights are chosen carefully, this Hopfield network is stable and converges to neuron activation values u^* which minimize the quadratic form

$$E = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} u_i^* W_{ij} u_j^* - \sum_{i=1}^{n} V_i u_i^*. \tag{2}$$

B. Optimal Control Problem

The synthesis of an optimal controller using a Hopfield neural network is presented to illustrate the feasibility of using a neural network for the missile guidance problem. This involves posing the missile pursuit as an LQG control problem and manipulating it into the Hopfield network form of (2). Consider minimizing the functional

$$J = S_{R_f}^T S_{R_f} + \gamma \int_{t_0}^{t_f} u^T u \, dt \tag{3}$$

with the differential equations of motion of the *relative* dynamics of the target and the missile given by

$$\dot{S}_R = V_R$$
$$\dot{V}_R = A_T - A_M = A_T - u \tag{4}$$

where

$$S_R = [X_T - X_M,\ Y_T - Y_M]^T$$
$$V_R = [\dot{X}_T - \dot{X}_M,\ \dot{Y}_T - \dot{Y}_M]^T \tag{5}$$

and

$$A_T = [A_{T_x}, A_{T_y}]^T, \qquad A_M = [A_{M_x}, A_{M_y}]^T. \tag{6}$$

In these equations, γ is the weight penalty on the control, X_T, Y_T are components of the target positions, X_M, Y_M are components of the missile positions, A_{T_x}, A_{T_y}, A_{M_x}, A_{M_y} are components of the accelerations of the target and the missile in x and y directions, respectively, and $(\dot{\cdot})$ denotes differentiations with respect to time. The first term of (3) forces the range distance S_R to be zero at the final time which gives the intercept. The second term minimizes the missile acceleration which is directly related to the amount of propellant expended.

The final time t_f is computed as

$$t_f = \frac{R}{|\dot{R}|} \tag{7}$$

where R is the relative range and $\dot{R}$ is the range rate at time t.

C. Hopfield Network for Optimal Control

In order to solve this optimal control problem using the Hopfield neural network, the performance index J must be formulated as a Hopfield energy function J_N in the form of (2) so that the network will converge to a fixed point. This fixed point represents the value of optimal control. In this section, the necessary derivation of such a form called the energy function for the Hopfield network will be carried out.

We can write the final distance, S_{R_f} as

$$S_{R_f} = S_{R_0} + V_{R_0}(t_f - t_0) - \int_{t_0}^{t_f}\int_{t_0}^{t}[u(\tau) - A_T(\tau)]\,d\tau\,dt. \tag{8}$$

The following quantities are now defined:

$$\Delta t = (t_f - t_0)/N$$
$$u_\alpha = u(t_0 + \alpha\Delta t), \qquad \alpha = 0,1,\ldots,N \tag{9}$$
$$a_\alpha = A_T(t_0 + \alpha\Delta t), \qquad \alpha = 0,1,\ldots,N$$

where N is a positive integer greater than zero.

With the assumptions that the Riemann integrals in J exist, we rewrite S_{R_f} as

$$S_{R_f} = -\lim_{N\to\infty}\sum_{i=1}^{N}\sum_{\alpha=1}^{i}(u_\alpha - a_\alpha)(\Delta t)^2 + S_{R_0} + V_{R_0}N\Delta t. \tag{10}$$

As a result, the product $S_{R_f}^T S_{R_f}$ is

$$\begin{aligned} S_{R_f}^T S_{R_f} &= \lim_{N\to\infty}\sum_{i,j=1}^{N}\sum_{\alpha=1}^{i}\sum_{\beta=1}^{j}(u_\alpha - a_\alpha)^T(u_\beta - a_\beta)(\Delta t)^4 \\ &\quad - \lim_{N\to\infty}(S_{R_0} + V_{R_0}\Delta t N)^T\sum_{i=1}^{N}\sum_{\alpha=1}^{i}(u_\alpha - a_\alpha)(\Delta t)^2 \\ &\quad - \lim_{N\to\infty}\sum_{i=1}^{N}\sum_{\alpha=1}^{i}(u_\alpha - a_\alpha)^T(\Delta t)^2(S_{R_0} + V_{R_0}\Delta t N) \\ &\quad + (S_{R_0} + V_{R_0}\Delta t N)^T(S_{R_0} + V_{R_0}\Delta t N). \end{aligned} \tag{11}$$

Similarly, we can rewrite the integral term in the performance index as

$$\begin{aligned} \gamma\int_{t_0}^{t_f} u^T u\,dt &= \lim_{N\to\infty}\gamma\sum_{\beta=1}^{N}u_\beta^T u_\beta\Delta t \\ &= \lim_{N\to\infty}\gamma\Delta t\sum_{\alpha,\beta=1}^{N}u_\alpha^T I_{\alpha\beta}u_\beta \end{aligned} \tag{12}$$

where

$$I_{\alpha\beta} = \begin{cases} 0 & \text{if } \alpha \neq \beta \\ I_{2\times 2} & \text{if } \alpha \le i\ \beta \le j \end{cases}$$
$$I_{2\times 2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{13}$$

We also define

$$T_{\alpha\beta}^{ij} = \begin{cases} 0 & \text{if } \alpha > i \ \ \vee\ \beta > j \\ I_{2\times 2} & \text{if } \alpha \le i \ \ \beta \le j \end{cases}. \tag{14}$$

With the use of these equations, we can define a performance index J_N as

$$\begin{aligned} J_N &= (\gamma\Delta t)\sum_{\alpha,\beta=1}^{N}u_\alpha^T I_{\alpha\beta}u_\beta \\ &\quad + (\Delta t)^4\sum_{i,j,\alpha,\beta=1}^{N}[u_\alpha^T T_{\alpha\beta}^{ij}u_\beta - a_\alpha^T T_{\alpha\beta}^{ij}u_\beta - u_\alpha^T T_{\alpha\beta}^{ij}a_\beta] \\ &\quad - (\Delta t)^2(S_{R_0} + V_{R_0}\Delta t N)^T\sum_{i,\alpha=1}^{N}T_{\alpha 1}^{il}u_\alpha \\ &\quad - (\Delta t)^2\sum_{i,\alpha=1}^{N}u_\alpha^T T_{\alpha 1}^{il}(S_{R_0} + V_{R_0}\Delta t N). \end{aligned} \tag{15}$$

Note that the terms independent of u have not been included. In order to place J_N in a final energy function form that can be used with Hopfield neural network optimization, we define $W_{\alpha\beta}$ and V_α. These are

$$W_{\alpha\beta} = -2\gamma(\Delta t)I_{\alpha\beta} - 2(\Delta t)^4 \sum_{i,j=1}^{N} T_{\alpha\beta}^{ij} \tag{16}$$

and

$$V_\alpha = (\Delta t)^4 \sum_{i,j,\beta=1}^{N} [a_\beta^T T_{\beta\alpha}^{ij} + T_{\alpha\beta}^{ij} a_\beta] + (\Delta t)^2 (S_{R_0} + V_{R_0}\Delta t N)^T \sum_{i=1}^{N} T_{\alpha 1}^{il} + (\Delta t)^2 \sum_{i=1}^{N} T_{\alpha 1}^{il}(S_{R_0} + V_{R_0}\Delta t N). \tag{17}$$

With these variables, the performance index can be cast in a Hopfield form as

$$J_N = -\frac{1}{2}\sum_{\alpha,\beta=1}^{N} u_\alpha^T W_{\alpha\beta} u_\beta - \sum_{\alpha=1}^{N} V_\alpha u_\alpha. \tag{18}$$

Now the optimal control problem is to find u_α^* for each α, to minimize J_N.

Since the target maneuvers a_α are unknown a priori, current best-guess estimates are continuously determined and used for these values. The optimal control problem is solved at each time instant during the actual pursuit, with time t varying from the current time t_0 to the current estimate for the intercept time t_f and with the initial parameters, range S_{R0} and range rate V_{R0} set to the current values of the target and missile. Herein lies the advantage of formulating the optimal control problem as a Hopfield network. In order to continuously solve and update the control problem in real time during the pursuit and intercept a very fast computational device is required. Tank and Hopfield [13] have shown that analog circuits can be constructed which perform a "massively parallel" implementation of a minimization of J_N. In addition, other hardware is available which implement Hopfield type networks at the speeds required.

D. Convergence of Hopfield Network

It is important to ensure the convergence of the network at all times during any scenario. The network weights are shown to be bounded, and if the time step is chosen small enough, then by the use of a contraction mapping theorem from point set topology, the network is shown to converge to a stable fixed point [16, 17].

THEOREM *For a fully connected recurrent network composed of n artificial neurons with activation dynamics*

$$u_i^{t+\Delta t} = \sum_{j=1}^{n} W_{ij} f(u_j^t) + V_i \tag{19}$$

where f is a bounded, continuous, differentiable, and real-valued function on the real line with bounded derivative; if

$$|f'_{\max}||w_{ij}| < C^* < \frac{1}{n} \tag{20}$$

for all i,j then the network converges to a unique fixed point for any initial conditions.

PROOF. Let Φ map R^n to R^n such that

$$\Phi(u) = \{y_i\}_{i=1}^n \qquad y_i = \sum_{j=1}^{n} W_{ij} f(u_j) + V_i. \tag{21}$$

Then the network of (19) can be written as $u^{t+\Delta t} = \Phi(u^t)$. The product space R^n is complete with the topology of the metric

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i|, \qquad x,y \in R_n. \tag{22}$$

The idea here is to show that Φ is a contraction mapping so that there exists a unique fixed point by the use of contraction mapping theorem [18].

For arbitrary choices of x, z in R^n

$$d(\Phi(x),\Phi(z)) = \sum_{i=1}^{n}\left|\sum_{j=1}^{n} W_{ij} f(u_j) + V_i - \sum_{j=1}^{n} W_{ij} f(z_j) - V_i\right| \le \sum_{i=1}^{n}\sum_{j=1}^{n} |f'_{\max}||W_{ij}||x_j - z_j| \tag{23}$$

by using the Lemma below. If the condition of (20) holds then

$$d(\Phi(x),\Phi(z)) \le \left(\sum_{i=1}^{n} C^*\right)\sum_{j=1}^{n}|x_j - z_j| \le C\, d(x_j, z_j) \tag{24}$$

where

$$C = \sum_{i=1}^{n} C_* < 1. \tag{25}$$

By the contraction mapping theorem, the network mapping Φ has a single fixed point x^* such that $u^* = \phi(u^*)$.

LEMMA *For any bounded, continuous, differentiable and real valued function f on the real line, with bounded derivative:*

$$|f(x) - f(y)| \le |f'_{\max}||x - y|, \qquad \textit{for all} \quad x,y \in R. \tag{26}$$

PROOF. This is a direct result of the Law of the Mean [19]. Let x, y belong to R. Then by the law of the mean, there exists a z in $[x,y]$ such that

$$f'(z) = \frac{f(x)-f(y)}{x-y}. \tag{27}$$

That is,

$$|f(x)-f(y)||f'(z)(x-y)| \le |f'_{\max}||x-y| \tag{28}$$

where

$$f'_{\max} = \text{limsup}|f'(x)|.$$

PROOF OF CONVERGENCE OF HOPFIELD NETWORK. In order to prove that the Hopfield network yields a converged solution at each point, note that the Hopfield network dynamics are governed by the equation

$$u_\alpha = \sum_{\beta=1}^{N} W_{\alpha\beta} f(u_\beta) + V_\alpha \tag{29}$$

where f is the sigmoid activation function. The weights of the Hopfield network are given by (16) as

$$W_{\alpha\beta} = -2\gamma(\Delta t)I_{\alpha\beta} - 2(\Delta t)^4 \sum_{i,j=1}^{N} T^{ij}_{\alpha\beta}. \tag{30}$$

By using (13) and (14), we can show that

$$-W_{\alpha\beta} \le 2\gamma(\Delta t)I_{2\times 2} + 2(\Delta t)^4 (N)^2 I_{2\times 2}. \tag{31}$$

If

$$\Delta t < \min\left\{ \frac{1/2}{2\gamma f'_{\max}(2N)}, \left(\frac{1/2}{2N^2 f'_{\max}(2N)}\right)^{1/4} \right\} \tag{32}$$

then,

$$-W_{\alpha\beta} < \frac{1/2}{f'_{\max}2N} I_{2\times 2} + \frac{1/2}{f'_{\max}(2N)} I_{2\times 2}. \tag{33}$$

That is,

$$-W_{\alpha\beta} < \frac{1}{f'_{\max}2N} I_{2\times 2}. \tag{34}$$

Since there are $2N$ neurons in the network, the magnitude of each scalar weight which is a component of $W_{\alpha\beta}$ is strictly bounded above by

$$\frac{1}{f'_{\max}2N} \tag{35}$$

then by the theorem above, the Hopfield network converges to a stable fixed point.

III. NUMERICAL EXPERIMENTS

Numerical results from using the Hopfield formulation for optimal control are presented in this section. The numerical results from a simple example

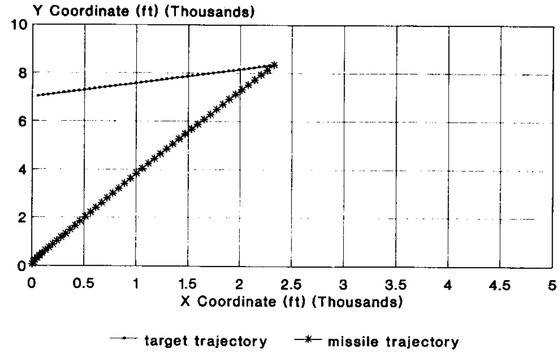

Fig. 1. Trajectories of missile and target. (AAY = 60°, BAY = 0°, range = 7000 ft.)

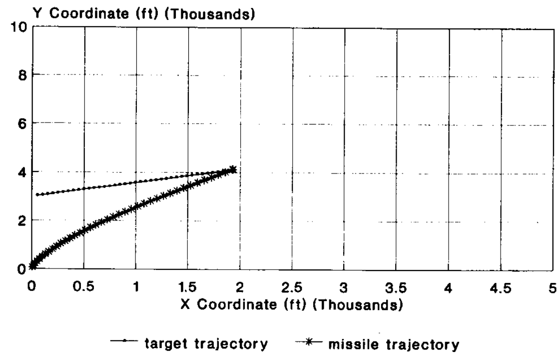

Fig. 2. Trajectories of missile and target. (AAY = 60°, BAY = 0°, range = 3000 ft.)

are presented in Table I. The initial conditions for the target have been assumed (in a normalized way) as 0.1 and 0.1 for positions and 1 and 0 for velocities, respectively. The target is assumed nonmaneuvering. The initial missile positions and velocities are zero. The output in Table I shows the time histories of the components of the relative range between the target and the missile. It can be observed that the missile intercepts the target. Engagement histories for various launch geometry of the target and the missile are also considered. The constraint velocity target and the missile are both assumed to have an initial Mach number of 0.9 at an altitude of 10000 ft. For initial ranges of 3000 ft and 7000 ft for different velocity directions, the missile and the target trajectories are presented in Figs. 1–5. It can be observed that the missile intercepts the target in all the cases considered. For more realistic cases, however, constraints on the capabilities of the missile have to be included. However, this should not present a problem in the utilization of the Hopfield neural network for use in guidance problems.

IV. CONCLUSIONS

A Hopfield neural network formulation of an optimal guidance problem is presented. The numerical

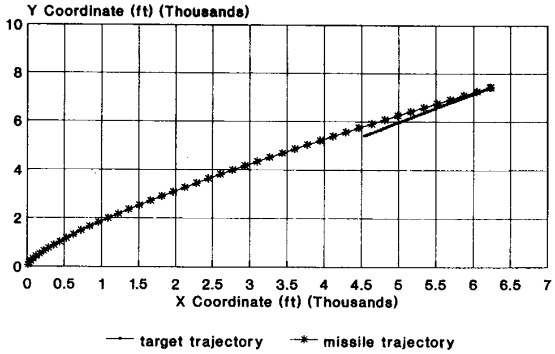

Fig. 3. Trajectories of missile and target. (AAY = 0°, BAY = 40°, range = 7000 ft.)

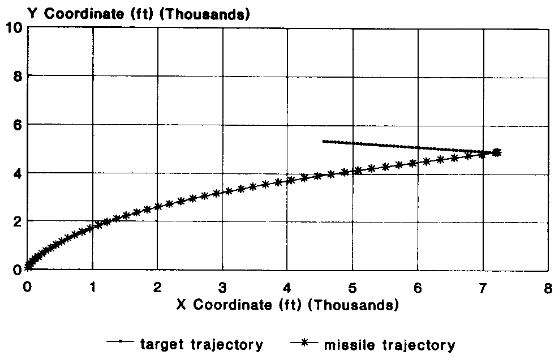

Fig. 4. Trajectories of missile and target. (AAY = 60°, BAY = 40°, range = 7000 ft.)

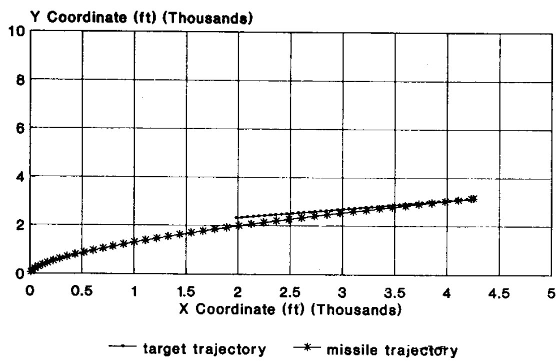

Fig. 5. Trajectories of missile and target. (AAY = 150°, BAY = 40°, range = 3000 ft.)

TABLE I
Histories of Relative Range Components with Time

Initial Target Position (.1,.1) Vel (.1,.0)
Target Acceleration = (.0,.0)
Initial Missile Position (.0,.0) Vel (.0,0)

	HOPFIELD	
TIME	XRANGE	YRANGE
0.05	0.09956	0.10402
0.10	0.09872	0.10714
0.15	0.09754	0.10944
0.20	0.09605	0.11102
0.25	0.09430	0.11192
0.30	0.09233	0.11224
0.35	0.09017	0.11202
0.40	0.08784	0.11133
0.45	0.08539	0.11022
0.50	0.08283	0.10875
0.55	0.08018	0.10696
0.60	0.07747	0.10489
0.65	0.07472	0.10259
0.70	0.07194	0.10008
0.75	0.06915	0.09742
0.80	0.06637	0.09461
0.85	0.06360	0.09170
0.90	0.06085	0.08871
0.95	0.05814	0.08565
1.00	0.05547	0.08256
1.05	0.05283	0.07942
1.10	0.05024	0.07625
1.15	0.04769	0.07305
1.20	0.04516	0.06981
1.25	0.04268	0.06654
1.30	0.04022	0.06324
1.35	0.03779	0.05991
1.40	0.03539	0.05656
1.45	0.03302	0.05318
1.50	0.03066	0.04978
1.55	0.02833	0.04636
1.60	0.02602	0.04292
1.65	0.02373	0.03946
1.70	0.02145	0.03599
1.75	0.01918	0.03251
1.80	0.01692	0.02902
1.85	0.01466	0.02553
1.90	0.01241	0.02204
1.95	0.01015	0.01855
2.00	0.00789	0.01505
2.05	0.00564	0.01156
2.10	0.00338	0.00807
2.15	0.00112	0.00458
2.20	-0.00113	0.00109

results indicate that this method is promising. It should be noted that in the Cartesian coordinate system, a conventional LQG formulation gives an explicit control law. However, many studies have brought out the superiority of the polar coordinates where measurements are linear functions of states. In such formulations, if the nonlinear equations of motion are used, the control law will not be explicit and will involve iterative computations. In such cases, Hopfield nets can be a good alternative. Since the neural networks involve parallel computing, effective use of computer time is an additional incentive in its implementation.

JAMES E. STECK
Department of Mechanical Engineering
The Wichita State University
Wichita, KS 67260-0035

S. N. BALAKRISHNAN
Department of Mechanical and Aerospace Engineering and Engineering Mechanics
University of Missouri-Rolla
Rolla, MO 65401

REFERENCES

[1] Adler, F. P. (1956)
Missile guidance by three-dimensional proportional navigation.
Journal of Applied Physics, **27** (1956).

[2] Bryson, A. E., and Ho, Y. C. (1969)
Applied Optimal Control.
Waltham, MA: Blaisdell, 1969.

[3] Sammons, J. M., Balakrishnan, S. N., Speyer, J. L., and Hull, D. G. (1979)
Development and comparison of optimal filters.
Final report, AFATL-TR-79-87, Oct. 1979.

[4] Balakrishnan, S. N., and Speyer, J. L. (1986)
A coordinate transformation based on filter for improved target tracking.
Journal of Guidance, Control, and Dynamics (Nov.–Dec. 1986), 704–709.

[5] Balakrishnan, S. N., and Speyer, J. L. (1989)
Assumed density filter with application to homing missile guidance.
Journal of Guidance, Control, and Dynamics (Jan.–Feb. 1989), 4–12.

[6] Balakrishnan, S. N. (1989)
An extension to modified polar coordinates and application with passive measurements.
AIAA Journal of Guidance, Control, and Dynamics (Nov.–Dec. 1989), 906–912.

[7] Cloutier, J. R., Evers, J. H., and Feely, J. J. (1988)
An assessment of air-to-air missile guidance and control technology.
In *Proceedings of the 1988 American Control Conference*, Atlanta, GA, June 1988, 133–142.

[8] Kohonen, T. (1988)
An introduction to neural computing.
Neural Networks, **1** (1988), 3–16.

[9] Carpenter, G., and Grossberg, S. (1987)
A massively parallel architecture for a self-organizing neural pattern recognition machine.
Computer Vision, Graphics, and Image Processing, **37** (1987), 54–115.

[10] Dayhoff, J. E. (1990)
Neural Network Architectures, An Introduction.
New York: Van Nostrand Reinhold, 1990.

[11] Lippman, R. P. (1987)
An introduction to computing with neural nets.
IEEE Transactions on Acoustics, Speech, and Signal Processing (Apr. 1987), 4–22.

[12] Hopfield, J. (1982)
Neural networks and physical systems with emergent collective computational abilities.
In *Proceedings of the National Academy of Science U.S.*, **79** (1982), 2554–2558.

[13] Tank, T., and Hopfield, J. (1986)
Simple neural optimization networks: An A/D converter, signal decision circuit, and a linear programming circuit.
IEEE Transactions on Circuits and Systems, **CAS-33**, 5 (May 1986), 533–554.

[14] Rumellhart, D. E., Hinton, G. E., and Williams, R. J. (1986)
Learning internal representations by error propagation.
In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. I.
Cambridge, MA: Bradford Books, 1986.

[15] White, D. A., Sofgee, D. A. (Eds.) (1992)
Handbook of Intelligent Control, Neural, Fuzzy, and Adaptive Approaches.
New York: Van Nostrand Reinhold, 1992.

[16] Steck, J. E. (1992)
Convergence of recurrent networks as contraction mappings.
In *Proceedings of the International Joint Conference on Neural Networks 1992*, Vol. III, 1992, 462–467.

[17] Li, L. K. (1992)
Fixed point analysis for discrete-time recurrent neural networks.
In *Proceedings of the International Joint Conference on Neural Networks 1992*, Vol. IV, 1992, 134–139.

[18] Greene, R. E. (1983)
Introduction to Topology.
New York: Saunders College Publishing, 1983, 43.

[19] Goldberg, R. R. (1976)
Methods of Real Analysis.
New York: Wiley, 1976, 203.

Errata "A Series Representation of the Spherical Error Probability Integral"

The following changes should be made to correct errors in [1]. The author apologizes for these inaccuracies.

In Eq. 5 and the immediately following equation, change $(z - z^{-1}/2i)^2$ to $((z - z^{-1})/2i)^2$.

In Eqs. 16 and 17, change $(4m + 2n + 1/2)$ to $(4m + 2n + 1)/2$.

In Eq. 19 delete the = to get the product of the constant term and the series

$$R^3 C_2 \sum_{m=0}^{\infty} \frac{\alpha^m}{(m!)^2} \sum_{n=0}^{\infty} \frac{\beta^n}{n!} \sum_{l=0}^{\infty} \frac{\xi^l}{l!}.$$

Between Eqs. 21 and 22 change a^m to α^m.

In Eq. 24 change e_α, e_β, e_ξ to $\epsilon_\alpha, \epsilon_\beta, \epsilon_\xi$ respectively.

In the equation for dF/dz following Eq. 28, change $(R^2 - z^2 - y^2/\sigma_x^2)$ to $(R^2 - z^2 - y^2)/\sigma_x^2$ in the exponential. Make the same change in the exponential in the inequality following the equation for dF/dz.

In the $\eta_{\min}$ equation following Eq. 29, change the product $4\surd$ to a fourth root to get

$$\eta_{\min} = \sqrt[4]{\frac{4\sigma_x \sigma_y}{9\sigma_z^2}}.$$

STEPHEN DEL MARCO
AWARE, Inc.
One Memorial Drive
Cambridge, MA 02142-1301

REFERENCES

[1] Del Marco, S. (1993)
A series representation of the spherical error probability integral.
IEEE Transactions on Aerospace and Electronics Systems, **29**, 4 (Oct. 1993), 1349–1356.

Manuscript received October 1, 1993.

IEEE Log No. T-AES/30/1/13946.

0018-9251/94/$4.00