Critique of some neural network architectures and claims for control and estimation

Article *in* IEEE Transactions on Aerospace and Electronic Systems · May 1998

DOI: 10.1109/7.670323 · Source: IEEE Xplore

1 author:

Thomas Henderson Kerr III
TeK Associates

Critique of Some Neural Network Architectures and Claims for Control and Estimation

THOMAS H. KERR, Senior Member, IEEE
TeK Associates

While there is great potential for successful use of neural network (NN) algorithms in automatic target recognition (ATR) and other pattern identification/classification applications, significant barriers have been encountered (as summarized herein) that, to date, defy rigorous use of NNs within feedback control designs. The status of several problems and contradictions involving NNs relating to control and estimation theory applications (and to practical failure detection within INS/GPS navigation systems) are summarized here. To give a positive spin and for a balanced perspective, we also mention many novel laudable NN results obtained by invoking the techniques and results of control and estimation theory.

Manuscript received September 2, 1995; revised October 24, 1996 and May 30, 1997.

IEEE Log No. T-AES/34/2/03187.

This work was supported by TeK Associates' IR&D Contract 95-105.

The work extends what appears in the *Proceedings of the International Conference on Signal Processing Applications and Technology*, Oct. 1995.

Author's address: TeK Associates, 9 Merriam St., Suite 7-R, P.O. Box 459, Lexington, MA 02173-0005.

0018-9251/98/$10.00

I. INTRODUCTION

While the present author prefers to engage in completely constructive investigations and development activities without conveying negative overtones, he occasionally finds it necessary to issue warnings to assist others along the path (e.g., [1–6, 121]). There is obviously great potential for successful use of neural network (NN) algorithms in automatic target recognition (ATR) [7, 15], speech recognition [8], and other pattern identification/classification applications [56–59] that *appropriately* match the underlying NN architectures. To date, there has been good symbiosis between NN and the methodology and techniques of control and estimation theory such as: 1) invoking Lyapunov functions to establish stability/convergence in NN learning [61–64], and 2) use of (extended) Kalman filters instead of backpropagation to accomplish NN learning in fewer iterations [37–39, 42, 59]; however, significant barriers have been encountered in going in the opposite direction of seeking to apply NN in areas of control and estimation. While NN critiques and NN control implementation surveys have appeared before [96, 97, 104] there is virtually no overlap with the new topics covered, questions raised, and insights offered here.

II. APPLYING HISTORICAL ADAPTIVE CONTROL RESULTS TO NN

Addressing individual contributions: Kumpati "Bob" S. Narendra resurrected his old sensitivity analysis technique from adaptive control (an area that he admits was previously somewhat heuristic prior to [34]) from 15 years ago and adapted it to understanding backpropagation learning in *feedforward* NNs (since it's similar to earlier gradient matrix techniques). From this particular specialist in nonlinear systems, results are always interesting and he always has proper explicit delimiters of applicability (somewhere) in the prose. While Narendra's four case representation [11, p. 12] is appealing and at first glance looks to be fairly general, encompassing not only purely linear structures (Case 1) but also nonlinear structures of three different kinds of increasing generality (Cases 2–4), it is noted from the examples considered in his paper that although having the initial appearance of total generality, they are in fact fairly restrictive and selected so that the denominator usually exceeds or matches the degree of the numerator asymptotically with time and the denominator structure is also such that no singular points[1] (regular or otherwise) are ever

[1]In comparison, the familiar Sturm-Louville (S-L) 2nd-order ODEs [47] that arise naturally in applying separation-of-variables to Maxwell's partial differential equations of EM theory result in the special functions of Bessel, Legendre, Hermite, etc., as solutions,

encountered. In fact the time domain evolution in each case is always so benign that the solutions will never be unbounded (as can be seen immediately without having to solve them).

Narendra states up front that his approach requires a *global* Lipschitz condition to be satisfied in order to be applicable. Recall that ordinary differential equations ODEs need satisfy merely a *local* Lipschitz condition to guarantee *existence and uniqueness* of solutions while, by way of comparison, stochastic differential equations (e.g., of Stratonovitch, Ito, or McShane type) historically must satisfy the more stringent *global* Lipschitz condition for existence and uniqueness of solutions (as required for a rigorous proof in the more challenging stochastic situation [106, p. 19] that also requires invocation of the Borel–Cantelli lemma [105, p. 42, expression 3.5.5] but which doesn't occur for the deterministic case) and so this global requirement constitutes a much, much narrower class of differential equations that one can rigorously obtain useful results for via Narendra's technique than is anticipated to exist for most practical applications. Yet this restrictive form is to be used just for "representation" of the applications' underlying mathematical model before proceeding to other goals of *identification, estimation, or control* using NNs (in other words, these later more standard goals are applicable using NNs only for systems that can be characterized as being of the form that fits within one of the four enumerated cases or categories that are declared to be a standard representation repertoire).

What if the actual system of interest doesn't fit nicely into the above four pigeon holes? Are all bets off in using NNs to handle it? How does this compare with prevalent techniques that invoke Weirstrass' or Kolmogorov's Approximation Theorem and claim NNs can accommodate or capture the essence of *any* type of nonlinearity (cf., [92])? Prof. Thomas Poggio (MIT) has been rigorously looking into the latter question (and into other questions) using Approximation Theory, splines, and functional analysis in splendid expositions [69–72].

Another second barrier to general engineering utility of Narendra's award winning paper is the likely non-applicability to the multi-output case. While plausible applicability to the multi-output case is demonstrated as a few special case examples, Narendra reveals that his primary analysis tool is feedback linearization of all nonlinearities present. The use of feedback linearization presumes: 1) *exact* knowledge of both the nonlinear structure and its parameters[2] with 2) no measurement noise being present, and 3) capability of instantaneous full access to *exact* control implementation variables (in order to exactly cancel any nonlinearities present). Each of the above three assumptions for applying input linearization appear to be questionable in actual practice,[3] as explained below.

Any transport delay incurred can aggravate the situation and interfere with successful cancellation, as would occur in trying to apply this technique to a real system. Narendra's usage is as follows. For a system described by a nonlinear ODE of the form

$$\dot{x}(t) = f(x,t) + G(x,t)u(t) \tag{1}$$

where $u(t)$ is a deterministic control to be user specified. The desired control is then taken to be

$$u(t) = G^{-1}(x,t)[-f(x,t) + Ax(t)] \tag{2}$$

for square invertible $G(x,t)$,[4] which when applied to the original system (i.e., (2) is substituted back into the original ODE of (1)), yielding

$$\dot{x}(t) = Ax(t) \tag{3}$$

for a well-behaved matrix A that the analyst specifies to suit his needs. Furthermore, (3) can be serviced with additional residual control as an additional term $Bu''(t)$ that can be included within the brackets of (2) to result in

$$\dot{x}(t) = Ax(t) + Bu''(t). \tag{4}$$

Thus (4) can then be manipulated to accomplish any stipulated goals, such as guiding the system to a designated location in state space as the target set to be attained at a specified time or to exhibit desirable behavior characteristics of overshoot and rise time (such as being critically damped as opposed to being under-damped or over-damped), or to possess desirable gain and phase margins for stability.

The technique *works in simulations* because one cancels precisely what one had originally modeled as the nonlinearity in the first place as it would have occurred in the simulation. However, for actual physical systems consisting of real analog components, the digitally implemented solution for cancellation is usually thwarted due to inaccuracies in machine representation, effects of quantization, truncation, and roundoff (and, more significantly, by likely transport delay since, in instrumented systems and processes, information about measured quantities doesn't appear instantly just where it is needed in

and each associated precursor ODE routinely possesses singularities in the real domain that make S-L challenging and useful to solve and their utility motivates why they are so extensively tabulated.

[2]My thanks to a reviewer who admonishes that two further presumptions under category 1 are 1′ that the inverse dynamics (nonlinear zero dynamics) must be stable, and 1″ that the relative degree must be known (but now being mitigated somewhat by the results of [115, 116]).

[3]It's somewhat amusing that certain authors warn about feedback linearization one place [30] yet embrace it elsewhere [94].

[4]In general systems usage outside of or predating NNs, there is no need for the control gain matrix $G(x,t)$ to be square or invertible and any nonlinear controllability present could still be demonstrated via the techniques of [117] when necessary.

computations but travels through wires, through relays, through hydraulic actuators, etc. and, as a consequence, incurs some finite delay[5]) such that received results almost never exactly match as being synchronized in time to the nonlinearities present in the physical system that one seeks to cancel in order to control the physical system. In seeking to apply the methodology of feedback linearization, the control to be used for actual systems would be of the form:

$$u(t) = G_0^{-1}(x(t-\tau) + w(t-\tau), t) \times [-f_0(x(t-\tau) + w(t-\tau), t) + A\{x(t-\tau) + w(t-\tau)\}] \tag{5}$$

but is an inexact match and wouldn't yield (3) as a result. Even if $f_0 = f$, $G_0 = G$ (other problems occur when G is not invertible [88]), and the noise effect were negligible as $w \equiv 0$, perfect cancellation *would still not occur* because of the delay term and the resulting differential-difference equation (being of the retarded type [17]) are subject to likely instabilities (just as one's bathroom shower water temperature adjustment attempt is unstable due to transport delay unless a predictive control strategy is employed that takes into account the slight time delay between what is currently sensed by the body and what will be sensed down stream in time as a result of making an adjustment) due to the magnitude of the delay. (Evidence of disappointing behavior being finally astonishingly concluded in [89] when no acceptable feedback control could be obtained using the techniques of [88] that correspond to above (1)–(4) at the heart of [11] for NN applications. Also see [95, 113] for other representative examples of feedback linearization.)

Furthermore, even in Narendra's ideal situation, if at least two controls are not present as effectively independent noninterfering controls in each of two channels for finagling with along his guidelines as extensions to basic feedback linearization, then all bets are off and Narendra's technique cannot be applied at all for even the two channel case (2 inputs/2 outputs) as the simplest multi-input/multi-output (MIMO) situation. More explicit substantiation behind this revelation is offered next since advocates claim that feedback linearization can be successfully applied in the MIMO case if the system exhibits the special "triangular" structure. We agree that their proposed technique can be applied to this particular type of MIMO system (if the issues of transport delay incurred and presence of sensor measurement noise are ignored or are negligible in a particular application) but we caution that these "triangular" systems are an extreme special case, as will be examined next to reveal just how "special" they are and whether any general transformations exist to convert arbitrary systems to be of this special triangular form.

Consider the state-variable representation of a general nonlinear system:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_n(t) \end{bmatrix} = \begin{bmatrix} f_1(x_1(t), x_2(t), \ldots, x_n(t), \mathbf{u}(t), t) \\ f_2(x_1(t), x_2(t), \ldots, x_n(t), \mathbf{u}(t), t) \\ \vdots \\ f_n(x_1(t), x_2(t), \ldots, x_n(t), \mathbf{u}(t), t) \end{bmatrix}. \tag{6}$$

Compare this with the considerably more restrictive structure of the following triangular systems (being the most general MIMO structure to which control linearization can always be applied):

$$\begin{bmatrix} \dot{x}'_1(t) \\ \dot{x}'_2(t) \\ \vdots \\ \dot{x}'_n(t) \end{bmatrix} = \begin{bmatrix} f'_1(x'_1(t), x'_2(t), \ldots, x'_n(t)) & & \\ 0 & f'_2(x'_2(t), \ldots, x'_n(t)) & \\ 0 & 0 & \vdots \\ 0 & 0 & f'_n(x'_n(t)) \end{bmatrix} + \begin{bmatrix} g'_1(t)u'_1(t) \\ g'_2(t)u'_2(t) \\ \vdots \\ g'_n(t)u'_n(t) \end{bmatrix} \tag{7}$$

where the single dot above a variable denotes a derivative with respect to time, while a primed function merely distinguishes it from the unprimed symbol as being a different function. The lure of the above nth-order triangular system is its tractability under feedback linearization, because it exhibits an *autonomous* (i.e., the $f'_i(\cdot)$ are not explicit functions of time t) upper triangular structure and additionally n independently specifiable linearly additive noninterfering controls (that no longer appear in the $f'_i(\cdot)$). Its tractable manipulations are analogous to those encountered in applying the back substitution step of *Gauss–Jordan* reduction or Gaussian elimination for matrices (but requiring further subsequent symbol manipulation algebra here). Starting with the last row, choose the control $u_n(\cdot)$ to cancel out the nonlinearity present as in this obvious generalization of (2),

$$u'_n(t) = \frac{1}{g'_n(t)}[-f'_n(x'_n(t), t) + A'_n x'_n(t) + B'_n u''_n(t)] \tag{8}$$

under the tacit assumption that each $g'_i(t) \neq 0$ for all $t > 0$, then the resulting linear ODE may be explicitly

[5]These comments are revelant as well in another context to control applications using a PC platform with M/S *Windows* 3.1x, /95, /NT operating systems since *Windows* messaging delays cause Comm traffic bottlenecks despite the presence of caching and processor independent 16-bit DMA busses and/or 32-bit PCI busses that don't drop or loose any data. However, there are remedies here such as use of a real-time *Windows*' OS with a pricetag that is an order of magnitude or two more expensive than routinely extracted by M/S.

solved for $x_n(t)$. This solution may be explicitly substituted in all the rows of (7) above this row and the ODE in the next row above may be solved for in like manner. Proceed up each row of (7) in the same way. Thus this "triangular" system can be handled or solved in *principle* and made to perform any task with the $u_i'(t)$s. (The details of actual handling is another story.)

To better understand the limitations or low likelihood of encountering real systems possessing or exhibiting this restrictive structure of (7) and to know how to handle them if they do possess it, consider the barriers exhibited by a more benign simpler merely linear time-varying versions of (6) and (7) being, respectively:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_n(t) \end{bmatrix} = \begin{bmatrix} a_{1,1}(t) & a_{1,2}(t) & \cdots & a_{1,n}(t) \\ a_{1,2}(t) & a_{2,2}(t) & \cdots & a_{2,n}(t) \\ \vdots & \vdots & \vdots & \vdots \\ a_{n,1}(t) & a_{n,2}(t) & \cdots & a_{n,n}(t) \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} + \begin{bmatrix} g_1(t)u_1(t) \\ g_2(t)u_2(t) \\ \vdots \\ g_n(t)u_n(t) \end{bmatrix} \quad (9)$$

and

$$\begin{bmatrix} \dot{x}_1'(t) \\ \dot{x}_2'(t) \\ \vdots \\ \vdots \\ \dot{x}_n'(t) \end{bmatrix} = \begin{bmatrix} a_{1,1}'(t) & a_{1,2}'(t) & \cdots & a_{1,n}'(t) \\ 0 & a_{2,2}'(t) & \cdots & a_{2,n}'(t) \\ 0 & 0 & \vdots & a_{3,n}'(t) \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & a_{n,n}'(t) \end{bmatrix} \begin{bmatrix} x_1'(t) \\ x_2'(t) \\ \vdots \\ \vdots \\ x_n'(t) \end{bmatrix} + \begin{bmatrix} g_1'(t)u_1'(t) \\ g_2'(t)u_2'(t) \\ \vdots \\ \vdots \\ g_n'(t)u_n'(t) \end{bmatrix}. \quad (10)$$

Again, (10) *appears* to be tractably handled by proceeding as in the *backwards substitution* step of Gauss–Jordan matrix reduction or Gaussian-elimination (familiar as arising in the machine solution of a system of linear constant coefficient algebraic equations). Starting from the bottom row of (10), this simple scalar ODE

$$\dot{x}_n'(t) = a_{n,n}'(t)x_n'(t) + g_n'(t)u_n'(t) \quad (11)$$

is recognized to have the following structural solution (by utilizing an integrating factor) of the form:

$$x_n'(t) = \exp\left[\int_0^t a_{n,n}'(\tau)\,d\tau\right] x_n'(0) + \int_0^t \exp\left[\int_0^\tau a_{n,n}'(w)\,dw\right] g_n'(\tau)u_n'(\tau)\,d\tau \quad (12)$$

which can be solved completely then explicitly substituted back into the next row up from the bottom as

$$\dot{x}_{n-1}'(t) = a_{n-1,n-1}'(t)x_{n-1}'(t) + \left[a_{n-1,n}'(t) \overbrace{x_n'(t)}^{\text{now known}} + g_{n-1}'(t)u_{n-1}'(t) \right] \quad (13)$$

which is recognized to be of the same fundamental form as (11) and so has a solution of a form similar to that of (12) as

$$x_{n-1}'(t) = \exp\left[\int_0^t a_{n-1,n-1}'(\tau)\,d\tau\right] x_{n-1}'(0) + \int_0^t \exp\left[\int_0^\tau a_{n-1,n-1}'(w)\,dw\right] \times [a_{n-1,n}'(\tau)x_n'(\tau) + g_{n-1}'(\tau)u_{n-1}'(\tau)]\,d\tau \quad (14)$$

where everything within the final brackets is a known specified function and consequently can be completely evaluated numerically. This same solution technique proceeds up the triangle, row by row (all of the same form), to the top row where it terminates naturally and the system has now been completely solved in principle. (Messy details are in how to properly convey the entire solution history of one row's $x_i'(t)$ to all the rows of (10) above it. A task that must be accomplished for $n-1$ $x_i'(t)$'s in $n-1$ rows!)

There was a decades long quest by engineers and mathematicians well aware of the potential highly lucrative payoff (in the 1960s [100–102, ch. 10]) to transform and convert (9) into (10) using a time-varying transformation as $x(t) = T(t)x'(t)$ (but still this aspect, related to so-designated *algebraic equivalence* and *topological equivalence* [101, p. 157] unfortunately came to no real consequence [103] although *strict equivalence*, for the time-invariant similarity transformation $x(t) = Tx'(t)$ [101, p. 158] did and resoundingly so). Taking just the benign linear time-varying triangular system of (10) to practical fruition using the approach described above has proved to be impossibly complex and daunting, despite the fact that it is seen to be possible theoretically. The challenge of successfully handling nonlinear triangular systems is no less daunting and the outlook no less bleak. No conversion or transformation path from general nonlinear system (6) to triangular (7) appears likely.[6] Equations 7 to 10 can be solved numerically by conventional Runge–Kutta predictor-corrector techniques for calculating solutions but can't be easily manipulated as an obvious strategy for control using the techniques that are the topic of this section except via heroic efforts for special cases of extremely low dimension and low complexity. Perhaps the situation is improving since [107, p. 6, col. 1] states that his "paper presents an architecture for on-line adaptive control that employs a neural network to compensate for inversion error present when feedback linearization methods are employed (by prior NNs) to control a dynamic process."

New bothersome worry is that [13] concludes "that ARMA[7] representation of systems in the frequency domain is more useful to achieve control goals in NN applications than time domain state variable representation." The historical trend has been the opposite: 1) the Wiener filter in the frequency domain, tractable for just the scalar single channel time-invariant linear case, gave way to the state variable-based Kalman filter (for optimal linear estimation), formulated in the time domain and tractable for MIMO and linear time-varying nonstationary situations; 2) the Burg *Maximum Entropy* algorithm, being an exact spectral estimator for only single channel situations and being merely an approximate linear estimation scheme for the multichannel MIMO case, required recasting in state variable form for Toeplitz matrices to emerge before applying Levinson–Durbin recursions in obtaining a multichannel spectral estimate. A contest between ARMA or state variable representation should be a moot point for the linear time-invariant situation considering the simple transformations that can be used to get between the two representations [9, p. 31, ex. 2.14] (cf. [10, eqs. 82, 109] for accomplishing the same type of conversion).

The above examples are just the tip of the iceberg relating to incompatibilities between use of neural networks for real world control and estimation applications. Many other examples will be given of open questions that need to be resolved to enable practical application of NN to control and estimation situations.

[6]Historically, other variants of these "triangular" systems have appeared (such as in the frequency domain or Laplace domain for transformed versions of (10) with all the $a_i^j(t)$s being polynomials in t but with closed-form solution similarly stymied). I thank the G.E. statistician, Dr. Paul Feder, for introducing me to them in 1971.

[7]A word of caution is that two different names have appeared to describe the same notion of controlled auto-regressive moving average (CARMA) in [96] and ARMAX by most statisticians and engineers specializing in the related area of parameter identification, where the X represents an eXogenous input. One of the reviewers believes that [118] offers a way forward in this area by availing a procedure to interpolate between multiple linear ARMAX models to better approximate a nonlinear ARMAX model by optimizing a global cost criterion.

III. NN APPROACHES TO FAILURE DETECTION IN DYNAMIC SYSTEMS

We have provided occasional surveys of the status of failure detection technology (elucidating the various emerging approaches) on several previous occasions (e.g., [5, 6, 16]) in keeping abreast of this fast changing area. We have developed first hand, identified, specified, or recommended preferred implementations for particular application situations or scenarios including that of detecting anomalous behavior of new navigation systems introduced on submarines (for details, see TASC reports TR-418-20, TR-512-3-1, TR-678-3-1, dated 1974, 1975, 1976) and for a multisensor navigation filter and failure detection, identification, and reconfiguration (FDIR) strategy in the Advanced Tactical Fighter (ATF) [14, 16].

While not desiring to be a "nay-sayer," the author has strong reservations about [18] along the lines that $E[f(x)]$ is not the same as $E[f(\hat{x})]$ when x is non-Gaussian or when $f(\cdot)$ is nonlinear[8] (as it is in [18]) and, moreover, is unknown (but bandied about later within [18] as if it were in fact known despite specifically acknowledging it to be unknown following [18, eq. 8] so that their eqs. 6, 7, and 8 are useless in their subsequent analytic endeavors). Shell games like this shouldn't occur. A ray of hope for using NN in inertial navigation system (INS) failure detection is exhibited in the evolving results of [60] supported by classical analysis [111, 112] and not deferred to the mysteries of NN learning.

While I am flattered that my prior failure detection algorithms [19] were deemed worthy of NN cloning, as cited in [109, 110], it alerted me to yet another area of concern in using NNs in discriminant analysis, pattern identification, and fault/failure detection. Namely, that NNs can be productively used as an alternative computational path to provide a test statistic, but the proper setting of the decision threshold level to correspond to prescribed consequential false alarm rates appears to still require the diligence of human analysis interceding as performed for [19] in [20, 21]. Just how one would proceed to specify a constant false alarm rate (CFAR) implementation for failure detection (or for radar or for any practical detection application) without a detailed human analysis to analytically specify the expressions for false alarm and correct detection eludes this author other than by massive experimental trials in an attempt to sort out receiver operating characteristic (ROC) curves which vary

[8]These lessons were learned in the early days of nonlinear filtering but are evidently still being forgotten (since there was no acknowledgement of a possible approximation being invoked here). In the notation of the author of [18], expectation is denoted by a carat above the symbol as in $\hat{x}$ rather than by the symbol $E[\cdot]$, as used here.

with intensity level of the signal being sought and naturally the intensity of background noise (as an effective signal-to-noise ratio (SNR)). Lacking a supporting rational analytic basis for NN decision inference (with guaranteed levels of false alarm rates and correct detection) does not automatically flow from just a collection of test cases provided for NN training, especially if noise is present (as always arises in trying to insert NNs into the practical application scenario). Similarly, to enable a CFAR detection for radar applications requires prescribing a varying decision threshold to maintain the false alarm rate at the specified constant level, where, apparently, all intermediate steps must be analytically specified explicitly.

IV. THE LURE AND LORE OF NN FOR CONTROL AND ESTIMATION

A. NN Lore

Gentle introductions to NNs are offered in [22, 23] regarding terminology, viewpoint, history, progress, and results. In particular, the utility of Lyapunov functions, familiar in control theory, and its important theoretical generalizations by Cohen and Grossberg [62], and Kosko [61] for NN that were crucial to guarantee stability/convergence and which influenced the development of NN architectures are highlighted in an extremely accessible form on [22, sec. 3.7, pp. 20–22]. An overview of NN from a control theorist's viewpoint is offered in [24] and further control theory ramifications are addressed in [26]. Other applications of NN in control and synergisms are addressed within the two special NN issues of *IEEE Control System Magazine*, Vol. 10, No. 3, 1990 and in Vol. 12, No. 2, Apr. 1992 and in Vol. 78, Nos. 9, 10 of the *IEEE Proceedings* (Sept., Oct.). New handbooks have also appeared [29] as well as more thorough and pedagogically correct self-study textbooks [27].

B. Some Cautions and Concerns

While not meaning to single out any one person since it was somewhat representative of all the papers in the afternoon NN session of the 1991 *American Control Conference* in Boston, L. Gordon Kraft (Univ. of N.H.) has discussed *Shennandoah's* CMAC (Cerebellar Model Articulation Controller)[9] control applications [84] that are apparently also for exclusively *feedforward* applications. Some of his overhead transparencies at the 1991 American Control Conference (ACC) unfortunately tended to nudge a viewer into believing otherwise (leading to a false sense of security) but eventually Kraft admitted (under direct questioning) to exclusively feedforward applicability and that their overhead transparencies should have had an (absent) link present in the feedforward path and one (portrayed to be in the feedback path) removed to be correct. Use of feedback (or feedforward along with feedback) is a mainstay of control engineers [91]. Its importance having been recognized and appreciated since the 1940s as 1) the means to reduce sensitivity of a design solution to parameter changes due to component aging (applications frequently arising in vacuum tube and transistor amplifiers). Other well-known benefits of using feedback control solutions are, 2) reducing adverse effects of nonlinearities and distortion, 3) providing an increased effective bandwidth or range of frequencies over which the system yields a satisfactory response or behavior, 4) increased accuracy for tracking or input-following (in faithfully reproducing a scaled version of it at the output). Perhaps other NN architectures such as Hopfield nets or other recurrent NNs [82, pp. 30–33], [113] can successfully accommodate this necessary feedback aspect usually needed in control for long term stability and robustness under varying conditions.

In too many NN applications addressing control, block diagrams appear to be somewhat contorted and shenanigans are played with connecting lines from one output device to a subsequent input. These shenanigans defy clarity of intent and vigilance has to be increased to ferret out the intent in such input/output block diagrams that were so clear for the past three decades before being abused this way. Evidently more than a few NN researchers looking into control related applications need to be watched fairly closely regarding what they convey in their figures and slides (even though they freely admit only feedforward applicability in the prose) because many busy viewers focus on these possibly misleading diagrams during public presentations at conferences instead of reading the disclaimers in the prose and, unfortunately, are lead to initially infer greater applicability to the feedback situation than has yet been rigorously established. As this article goes into review, a clear (planar) feedback diagram (without kinks) for control using NN was finally encountered in [90, Fig. 5, p. 734] by Lockheed[10] for lateral-directional control for high performance aircraft.

C. The Lure of NN

Some of the exciting new applications being pursued using NNs are: 1) classification of correlation

[9]Many NN practitioners characterize CMAC as a glorified table look-up routine.

[10]An interesting outcome since a prior internally funded Lockheed IR&D project [104] had pointed out perceived weaknesses in prior perceptron/backpropagation applications. However, Lockheed rejoined the NN fray in [114].

signatures of spread spectrum signals, 2) speech recognition, 3) image processing, 4) characterization of radar signals, 5) ATR, 6) implementing dynamic programming (DP)/Viterbi algorithm (DP $\equiv$ VA), 7) fault detection/diagnosis,[11] 8) multitarget tracking.[12]

Two NN aspects that may further interest a control theorist are: 1) use of Kalman filters and/or extended Kalman filters of NN as a more expedient alternative (requiring fewer iterations to convergence[13]) to the use of back-propagation for setting the weights (i.e., learning) in multilayer Perceptron type NNs, 2) utility of sophisticated Lyapunov functions for guaranteed NN convergence and its interaction in allowing the specification or design of candidate NN architectures that are likely to be fruitful by having theoretically guaranteed convergence properties. Another lure of NN use is in circumventing the need to totally specify the detailed mathematical model for the physical system beforehand but to merely allow the NN to adaptively learn on-line what the control actions should be to elicit the desired system response. Thus, the NN could successfully accommodate a changing environment or aging components with characteristics that deviate from the original status quo.

There is recent interplay between use of NN for control (handling/implementing sliding mode control) and use of control-based techniques to expedite NN learning (via use of Kalman filters and/or extended Kalman filters instead of back propagation, as mentioned above). A pressing question, touched on in Section IVB, is whether NNs can actually be used to implement feedback control, as is highly desirable from a practical point of view, but which to date appears to defy rigorous theoretical justification or even experimental confirmation (while reputable researchers, like Jim Spall [JHU/APL], reported at the 1991 Boston ACC [85] on his attempt at using NN feedback control using a type of *Stochastic Approximation*, it appeared to be too preliminary and not yet convincing).[14] Negative results were also reported at [45] by Henry Jex (Systems Technology, Inc.) regarding use of an Adaptive Clustering NN for feedback flight control of aircraft experiencing battle damage and seeking reconfiguration that augmented vectored thrusting and sought fail-safe limp home capability. The technique of [46] is somewhat ambiguous about whether the NN is feedback or feedforward in implementing an ART-based adaptive pole placement neurocontroller. The compatibility of NN implementations with feedback control will undoubtedly be clarified in future investigations.

From control theory we know that there is strong duality (as established by Kalman in 1959) between optimal regulation and optimal estimation (i.e., both applications possess similar quadratic cost functions for application situations involving an underlying linear system with positive definite weighting matrices appearing where needed and possessing a similar solution methodology involving a matrix Riccati equation because both problems have identical underlying mathematical structure[15]). Notable recent applications of basic control theory within NN technology has been as follows.

Realization that problem of specifying appropriate NN weights is a dynamic programming problem, as explicitly posed in [31],[16] with simplifying approximations likely [32];

Resurrection of sensitivity analysis approach (previously developed 20+ years ago in model adaptive control in using a gradient descent optimization[17] approach even though it lacked explicit analytical guarantee of convergence) to also be applicable to NN learning via so-designated *dynamic* back propagation [12] [also without analytic convergence guarantees] (cf., [35, 43]);

Recent revelation in [41] that techniques of classical optimal control, using familiar Hamiltonian formulation, can be adapted to serve as an overview for gauging the utility of differing learning approaches and even for suggesting other approaches to learning to expedite convergence as well as providing a unifying perspective for common understanding since several special cases are *outer product rule*, *recurrent back propagation*, and *spectral methods*;

[11] Some remedies to bad NN performance in this application area have been offered [52, 53].

[12] An overview of this area is [44], with new ideas expressed for use of NN within its pursuits.

[13] But perhaps with comparable operations counts [39] so its advantages are questionable.

[14] See follow-up work by Prof. Hanfu Chen, Chinese Academy of Sciences, visiting UC Santa Barbara and University of Kansas.

[15] A distinction is that Kalman filter Riccati Equation is to be solved forwards in time while that of the quadratic feedback regulator is to be solved backwards in time.

[16] The reverse has also occurred here where a NN has been proposed to provide approximate solution to dynamic programming problems [40] similar to those (Traveling Salesman Problem) initially investigated by Hopfield/Tank and criticized by Wilson/Pawley. We can't help but notice that in the same issue of the same journal, one paper says that NN learning in perceptrons (hidden Markov models) is a fundamental dynamic programming (DP) problem, while the other paper says that perceptron NNs can solve DP problems **(a classic Round Robin or vicious cycle)** but mitigated by the disclaimer or limitation that the type of DP claimed to be solvable by perceptron NNs are less difficult than the DP problem claimed to be equivalent to perceptron learning.

[17] The recent approach of [55] avoids gradient-based learning and as such exhibits great promise as well as exhibiting novel combinations of expert systems [54]. Researchers at University of Texas (Austin) are looking into similar implementations (1996) that involve a hierarchy of Kalman filters (treated as the experts) gated by NN-like structures and also similarly invoke the EM (Expectation-Maximization) algorithm.

Concepts of structural stability being exploited to gauge utility of unsupervised learning (ABAM, RABAM[18]) in feedback NN [36];

Use of extended Kalman filter instead of back propagation (demonstrated to be an extreme degenerate form of extended Kalman filter) for implementing a learning algorithm for a multilayered perceptron to set the weights [37–39] (but admitting that while fewer iterations were required, the overall operations counts were comparable and sometimes more for the extended Kalman filter). Reference [42] was able to avoid use of an extended Kalman filter entirely and instead use a much simpler standard linear Kalman filter to expedite learning in layered NN where it is noted that the effective learning rate is adaptive rather than constant as it is for back propagation and this variable rate is claimed to be an advantage in an auto-associative image compression application;

For multitarget tracking, typically the domain for use of Kalman filters, (M. Caudill and C. Butler, TRW Mead, 1987) investigate using Grossberg–Mingolla boundary contour system (BCS) NN to correlate sequences of target reports. BCS is interpreted as an interpolative probability field (IPF) (covering the region of space scanned by the sensor) with the impressive claim that the resulting multitarget tracking algorithm has a performance that is *independent of the number of objects tracked.* (Comment: usually, conventional multitarget trackers get saturated by having a computational architecture and fixed maximum track-file size and report association tree structure that must be searched (with increasing computational delay) for scan-to-scan data correlation continuity of same targets tracked, a computer burden that blossoms exponentially (NP-complete). Compare sparsely documented abstract NN approach to recent novel concrete advances made using explicit conventional analysis techniques in [119] that abstract NN studies siphon funds from[19]);

Multitarget tracking (E. Barnard and D. P. Casasent, Carnegie-Mellon, 1989) optical NN inference processor, NN imaging spectrometer, and NN matrix inversion device are described with initial results provided. Extensions made to Kalman filter implementation although its an energy-minimization approach;

Multitarget track initiation problem claimed to be solved (M. Lemmon, Carnegie-Mellon, 1989) by NN that is self-organizing and dominated by self-inhibition. The claim is that the NN will eventually cluster its internal states about the modes of a stimulating source and as a consequence can be used for parameter identification in situations characterized by non-Gaussian or other multimodal densities. (Comment: this situation exists in nonlinear filtering target tracking applications as well);

Problem of missing observations or missing measurement data treated using NN (M. Nakao and K. Hara and M. Kimura and R. Sato, Tohoku University, 1985). A counting process or point process of *intensity* is observed as the additional info for estimation. An extended Kalman filter is used for the estimation task. Algorithm is applied to artificial NN and to cat's visual nervous system;

Use of Hopfield NN probabalistic data association (NPDA) algorithm[20] to perform data association [50] within an application of multitarget tracking in clutter as an approximate recasting of the so-designated Joint Probabilistic Data Association (JPDA) scheme. (Comment: JPDA originally developed by Yaakov Bar-Shalom over the years (at University of Connecticut, previously at SCI/SCT) in conjunction with Tom Fortmann (BBN) and also by Dick Wishner's CA company (Edison Tse, Chong, Mori, etc.));

Novel and constructive invocation of additive white noise [65] (in a manner similar to what's done in random search for classical optimization) within feedback of Hopfield net to guarantee (via stochastic Lyapunov function [using martingales] a'la Harold Kushner) convergence of this stationary system (as a Gibb's field) to a prescribed statistical distribution ($1/Z \exp[-V(x)/T]$ having temperature T as a parameter) being a stationary Markov chain. By further lowering T (no faster than at a prescribed rate) interfaced these results with the *simulated annealing* framework with a Hopfield net so that global optimum is obtained instead of the mere local NN optima that are usually obtained. Use of randomization here invoked Stratonovich and Ito calculus within Hopfield network.

Except for the last item which is complete as it stands, it is acknowledged that most of the above are still being developed and refined.

D. Important Open Questions

Another advertised lure of NN use is in circumventing the need to totally specify the detailed mathematical model for the physical system beforehand but to merely allow the NN to adaptively learn on-line what the control actions should be to elicit the desired system response. Thus the NN could successfully accommodate a changing environment or aging components with characteristics that deviate from those originally present.

[18]An explicit counterexample to this approach has appeared relatively recently in [49].

[19]Perhaps this explains the recent request made by Paul J. Werbos at a public forum at MIT in April 1996 "for researchers to submit control and estimation proposals to him for possible NSF funding under the NN umbrella even if they don't have anything to do with Neural Networks per se". I fully expect the nice "Team Theory"results of fifteen years ago to resurface under the guise of NN.

[20]Drawbacks to the approach of [50] are offered in [51].

While the above is a noble goal, a more pressing sub-problem within this last category is how to go about distinguishing between problems or tasks that are tractably possible (using the standard NN trial-and-error approach) from those which are impossible to solve even with no limits on the time expended and number of trial tests performed other than it be finite. For example, two multilayered Perceptrons (with 26 nodes each) were used by Bernie Widrow (Stanford) [74–78] to solve the "truck backer-upper problem" and the less publicized "inverted pendulum a.k.a. the broom balancing problem on a mobile cart (attached by a pin)"[21] without furnishing an explicit model by instead allowing sufficient variety of experiments to be performed and the necessary processing take place to distill this learning (as associated CPU time expenditure) [83]. Two brooms, one on top of the other in tandem (attached by an additional pin), could be successfully handled similarly; however, two parallel brooms, with different offsets, both attached to the cart by the same original pin could never be handled because "controllability" is absent for this latter situation. In lieu of not consulting existing controllability analyses for this problem [66, pp. 10–17], how would an experienced NN practitioner unschooled in control theory know that the second case was impossible to solve due to physical principles, with or without a model being supplied to the NN and so shouldn't even be attempted without risking a waste of time and money in fruitless computations? Other attempts at applying NN for solving other problems could be just as hopeless as the last described broom balancing problem without any clue being available beforehand as a caution that it is impossible to solve and should instead be skipped or avoided entirely.[22]

Regarding priority in discovering the mechanization of NN backpropagation attributed to Paul J. Werbos in his 1974 Ph.D. thesis, [43] throws some light on earlier mechanizations of the gradient techniques used in this way by H. J. Kelley [67] and others (also see Raman Mehra's adaptive control and parameter identification techniques of the late 1960s and early 1970s [68] and Paul M. Frank's matrix sensitivity analysis [73] for control systems of even earlier vintage). After all, it merely consists of the chain rule from *Advanced Calculus*. Variations of an "Adaptive Critic"[23] (e.g., [108]) may possibly avoid "a need for an accurate atomic clock for distinguishing evaluations of the cost function $J(k)$ and $J(k-1)$ for comparisons between the current k and prior time $k-1$ in formulating improvements"[24] when utilization of an inexpensive tapped delay line or zero-order hold device will suffice instead.

A massive study of the efficacy of pursuing NN research (funded by DARPA) [86] concluded that NN implementations performed *twice as well* as conventional statistical pattern recognition techniques for doing the same thing. Later follow-up [87] (also funded by DARPA) clarified that the *performance is about the same* as that of conventional algorithms and mechanizations. [Both of the above conclusions were routinely revealed within a Boston Section IEEE NN Course lead by Dr. Beth Wilson (Raytheon).] It's little wonder that the historical perception of NN research to an outsider was that it was loaded with hype. Rank and file usually echo what's being broadcast at the highest leadership level. Also see [122].

REFERENCES

[1] a. Kerr, T. H. (1990)
Status of CR-like lower Bounds for nonlinear filtering.
IEEE Transactions on Aerospace and Electronic Systems, **25**, 5 (Sept. 1989), 590–601.

[1] b. Kerr, T. H. (1990)
Author's Reply.
IEEE Transactions on Aerospace and Electronic Systems, **26**, 5 (Sept. 1990), 896–898.

[2] Kerr, T. H. (1990)
Fallacies in computational testing of matrix positive definiteness/semidefiniteness.
IEEE Transactions on Aerospace and Electronic Systems, **26**, 2 (Mar. 1990), 415–421.

[3] Kerr, T. H. (1990)
On misstatements of the test for positive semidefinite matrices.
AIAA Journal of Guidance, Control, and Dynamics, **13**, 3 (May–June 1990), 571–572.

[4] Kerr, T. H. (1988)
Computational techniques for the matrix pseudoinverse in minimum variance reduced-order filtering and control.
In C. T. Leondes (Ed.), *Control and Dynamic Systems—Advances in Theory and Applications*, Vol. XXVIII: *Advances in Algorithms and Computational Techniques for Dynamic Control Systems*, Part 1 of 3. New York: Academic Press, 1988, 57–107.

[21] See [79–81] for history and other solution approaches to this historical 30+ year old (already solved) problem (related to the launch of multistage rockets as the practical application that motivates it).

[22] Prof. John Baillieul (Boston University) has a way of making both brooms stand up simultaneously (by vigorously vibrating the whole contraption vertically) but this is a different problem (since control is introduced in a different dimension orthogonal to that of the original problem) but definitely not practicable for multistage rockets.

[23] Possibly an "Adaptive Back-Seat Driver" while the main control is concerned about successfully driving down the road, the "Adaptive Critic" may be worried about other inconsequential issues as allowing cosmetics to be applied without smearing so that one is ready to meet the public when one arrives at the destination. I apologize for this sexist example but it makes the important point that care needs to be exercised to ensure that the two goals (which are simultaneous) are consistent on a detailed level as well.

[24] As directly quoted from Paul J. Werbos at 1992 Yale Workshop on *NNs and Adaptive Control.*

[5] Kerr, T. H. (1983)
Examining the controversy over the acceptability of SPRT and GLR techniques and other loose ends in failure detection.
In *Proceedings of the American Control Conference*, San Francisco, CA, June 1983, 906–977.

[6] Kerr, T. H. (1991)
Comments on "Federated Square Root Filter for Decentralized Parallel Processes."
IEEE Transactions on Aerospace and Electronic Systems, **AES-27**, 6 (Nov. 1991), 946–949.

[7] Kolodzy, P. J. (1993)
Multidimensional automatic target recognition system evaluation.
The Lincoln Laboratory Journal, **6**, 1 (1993), 117–145.

[8] Lippmann, R. P. (1987)
An introduction to computing with neural networks.
IEEE ASSP Magazine, **4**, 2 (Apr. 1987), 4–22.

[9] Chui, C. K., and Chen, G. (1987)
Kalman Filtering with Real-Time Applications.
New York: Springer-Verlag, 1987.

[10] Kerr, T. H. (1994)
Emulating random process target statistics (using MSF).
IEEE Transactions on Aerospace and Electronic Systems, **30**, 2 (Apr. 1994), 556–577.

[11] Narendra, K. S., and Parthasarathy, K. (1990)
Identification and control of dynamical systems using neural networks.
IEEE Transactions on Neural Networks, **1**, 1 (Mar. 1990), 4–27.

[12] Narendra, K. S., and Parthasarathy, K. (1991)
Gradient methods for the optimization of dynamical systems containing neural networks.
IEEE Transactions on Neural Networks, **2**, 2 (Mar. 1991), 252–262.

[13] Narendra, K. S., Mukhopadhyay, S., and Levin, A. U. (1992)
Adaptive control using neural networks.
In *Proceedings of Seventh Yale Workshop on Adaptive and Learning Systems*, Center for System Science, Durham Laboratory, New Haven, CT, May 20–22, 1992, 85–91.

[14] Carlson, N. A., Kerr, T. H., and Sacks, J. E. (1984)
Integrated navigation concept study.
Intermetrics Report IR-MA-321, June 15, 1984.

[15] Kerr, T. H. (1994)
Use of GPS/INS in the design of airborne multisensor data collection missions (for tuning NN-based ATR algorithms).
In Institute of Navigation *Proceedings of GPS-94*, Salt Lake City, UT, Sept. 20–23, 1994, 1173–1188.

[16] Kerr, T. H. (1987)
Decentralized filtering and redundancy management for multisensor navigation.
IEEE Transactions on Aerospace and Electronic Systems, **AES-23**, 1 (Jan. 1987), 83–119 (minor corrections appear on p. 412 of May and on p. 599 of July 1987 issues of same journal).

[17] Bellman, R., and Cooke, K. L. (1963)
Differential-Difference Equations.
New York: Academic Press, 1963.

[18] Farrell, J., Berger, T., and Appleby, B. (1993)
Using learning techniques to accommodate unanticipated faults.
IEEE Control Systems Magazine, **13**, 3 (June 1993), 40–49.

[19] Kerr, T. H. (1977)
Real-time failure detection: A static nonlinear optimization problem that yields a two ellipsoid overlap test.
Journal of Optimization Theory and Applications, **22**, 4 (Aug. 1977), 509–535.

[20] Kerr, T. H. (1980)
Statistical analysis of a two ellipsoid overlap test for real-time failure detection.
IEEE Transactions on Automatic Control, **AC-25**, 4 (Aug. 1980), 762–773.

[21] Kerr, T. H. (1982)
False alarm and correct detection probabilities over a time interval for restricted classes of failure detection algorithms.
IEEE Transactions on Information Theory, **IT-28**, 4 (July 1982), 619–631.

[22] Simpson, P. K. (1990)
Artificial Neural Networks: Foundations, Paradigms, Applications, and Implementations.
New York: Pergamon Press, 1990.

[23] Lawrence, J. (1992)
In S. Leudeking (Ed.), *Introduction to Neural Networks and Expert Systems* (4th ed.).
California Scientific Software, Grass Valley, CA, Jan. 1992.

[24] Melsa, P. J. W. (1989)
Neural networks: A conceptual overview.
Report TRC-89-08, Tellabs Research Center, Mishawaka, IN, Aug. 1989.

[25] Kosko, B. (1992)
Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence.
Englewood Cliffs, NJ: Prentice-Hall, 1992.

[26] Miller, W. T., Sutton, R. S., and Werbos, P. J. (1990)
Neural Networks for Control.
Cambridge, MA: MIT Press, 1990.

[27] Caudill, M., and Butler, C. (1992)
Understanding Neural Networks, Vols. 1 and 2 (with PC software disks).
Cambridge, MA: MIT Press, 1992.

[28] Korn, G. A. (1991)
Neural Network Experiments on Personal Computers and Workstations (with PC software disk).
Cambridge, MA: MIT Press, 1991.

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

[30] Slotine, J.-J. E., and Li, W. (1991)
Applied Nonlinear Control.
Englewood Cliffs, NJ: Prentice-Hall, 1991.

[31] Saratchandran, P. (1991)
Dynamic programming approach to optimal weight selection in multilayer neural networks.
IEEE Transactions on Neural Networks, **2**, 4 (July 1991), 465–466.

[32] Sartori, M. A., and Antsaklis, P. J. (1991)
A simple method to derive bounds on the size and to train multilayer neural networks.
IEEE Transactions on Neural Networks, **2**, 4 (July 1991), 467–470.

[33] Isermann, R., Lachmann, K.-H., and Matko, D. (1992)
Adaptive Control Systems.
Englewood Cliffs, NJ: Prentice-Hall, 1992.

[34] Ioannou, P. A., and Datta, A. (1991)
Robust adaptive control: A unified approach.
Proceedings of IEEE, **79**, 12 (Dec. 1991), 1736–1768.

[35] Zhao, Y., Bryson, A. E., and Slattery, R. (1990)
Generalized gradient algorithm for trajectory optimization.
AIAA Journal of Guidance, Control, and Dynamics, **13**, 6 (Nov.–Dec. 1990), 1166–1169.

[36] Kosko, B. (1991)
Structural stability of unsupervised learning in feedback neural networks.
IEEE Transactions on Automatic Control, **36**, 7 (July 1991), 785–792.

[37] Singhal, S., and Wu, L. (1989)
Training multilayer perceptrons with the extended Kalman algorithm.
In D. S. Turetzky (Ed.), *Advances in Neural Information Processing Systems I.*
San Mateo, CA: Morgan Kaufman, 1989, 113–140.

[38] Puskorius, G. V., and Feldkamp, L. A. (1991)
Decoupled extended Kalman filter training of feedforward layered networks.
In *Proceedings of the IJCNN*, **1** (July 1991), 771–777.

[39] Rogers, S. K., Ruck, D. W., Kabrisky, M., and Tarr, G. L. (1990)
AFIT neural network research.
IEEE AES Magazine, **5**, 9 (Sept. 1990), 17–19.

[40] Chiu, C., Maa, C.-Y., and Shanblatt, M. A. (1991)
Energy function analysis of dynamic programming neural networks.
IEEE Transactions on Neural Networks, **2**, 4 (July 1991), 418–426.

[41] Farotimi, O., Dembo, A., and Kailath, T. (1991)
A general weight matrix formulation using optimal control.
IEEE Transactions on Neural Networks, **2**, 3 (May 1991), 378–394.

[42] Huang, T., Tsuyuki, M., and Yasuhara, M. (1991)
A learning algorithm of the neural network based on Kalman filtering.
IEICE Transactions, **E-74**, 5 (May 1991), 1059–1065.

[43] Dreyfus, S. E. (1990)
Artificial neural networks, back propagation and the Kelley-Bryson gradient procedure.
AIAA Journal of Guidance, Control, and Dynamics, **13**, 5 (Oct. 1990), 926–928.

[44] Kim, H. K. (1991)
New filtering and data association algorithms for multiple target tracking.
Ph.D. dissertation, Dept. Electrical and Computer Engineering, Northeastern University, Boston, Aug. 29, 1991.

[45] In *Proceedings of Neural Network Applications Conference.*
Results of Phase 1 and 2 SBIRs.
Funded by Navy and Volpe National Transportation Systems Center, Park Tree Club, Crystal City, Arlington, VA, June 4–5, 1992.

[46] Kumar, S. S., and Guez, A. (1991)
ART based adaptive pole placement for neurocontrollers.
Neural Networks, **4** (1991), 319–335.

[47] Moon, P., and Spencer, D. E. (1961)
Field Theory for Engineers.
New York: Van Nostrand, 1961.

[48] Bello, M. G. (1992)
Enhanced training algorithms and integrated training/architecture selection for multilayer perceptron networks.
IEEE Transactions on Neural Networks, **3**, 6 (Nov. 1992), 864–875.

[49] Castanon, D. (1993)
A counterexample in stochastic stability of feedback neural networks.
IEEE Transactions on Automatic Control, **38**, 2 (Feb. 1993), 378–380.

[50] Sengupta, D., and Iltis, R. A. (1989)
Neural solution to the multiple target tracking data association problem.
IEEE Transactions on Aerospace and Electronic Systems, **25**, 1 (Jan. 1989), 96–108.

[51] Zhou, B., and Bose, N. K. (1993)
A comprehensive analysis of "Neural Solution to the Multitarget Tracking Data Association Problem".
IEEE Transactions on Aerospace and Electronic Systems, **29**, 1 (Jan. 1993), 260–263.

[52] Barnard, E., and Wessels, L. F. A. (1992)
Extrapolation and interpolation in neural network classifiers.
IEEE Control System Magazine (Oct. 1992), 50–53.

[53] Leonard, J. A., and Kramer, M. A. (1991)
Radial basis function networks for classifying process faults.
IEEE Control System Magazine (Apr. 1991), 31–38.

[54] Jacobs, R. A., and Jordan, M. I. (1993)
Learning piecewise control strategies in a modular neural network architecture.
IEEE Transactions on Man, Systems, and Cybernetics, **23** (Mar.–Apr. 1993), 337–345.

[55] Jordan, M. I., and Jacobs, R. A. (1994)
Hierarchical mixtures of experts and the EM algorithm.
Neural Computation, **6**, 2 (Mar. 1994), 181–214.

[56] Chin, L. (1994)
Applications of neural networks in target tracking data fusion.
IEEE Transactions on Aerospace and Electronic Systems, **30**, 1 (Jan. 1994), 281–287.

[57] Overman, T. L., and Louri, A. (1994)
A target detecting neural network architecture for serial sensor data.
In *Proceedings of IEEE Electro International*, Boston, MA, May 10–12, 1994, 393–403.

[58] Saerens, M. (1995)
Design of perception-like algorithms based on system identification techniques.
IEEE Transactions on Neural Networks, **6**, 2 (Mar. 1995), 504–506.

[59] Rao, K. D., and Sridhar, G. (1995)
Improving performance in pulse radar detection using neural networks.
IEEE Transactions on Aerospace and Electronic Systems, **31**, 1 (July 1995), 1193–1198.

[60] Da, R., and Lin, C.-F. (1995)
A failure diagnosis system using ARTMAP neural networks.
AIAA Journal of Guidance, Control, and Dynamics, **18**, 4 (July–Aug. 1995), 696–701.

[61] Kosko, B. (1988)
Feedback stability and unsupervised learning.
In *Proceedings of the IEEE International Conference on Neural Networks*, Vol. I, San Diego, CA, 1988, 141–152.

[62] Cohen, M., and Grossberg, S. (1983)
Absolute stability of global pattern formation and parallel storage by competitive neural networks.
IEEE Transactions on Systems, Man, and Cybernetics, **SMC-13** (1983), 815–825.

[63] Li, J.-H., and Michel, A. N. (1989)
Analysis and synthesis of a class of neural networks: Variable structure systems with infinite gains.
IEEE Transactions on Circuits and Systems, **36**, 5 (May 1989), 713–732.

[64] Grujic, L. T., and Michel, A. N. (1991)
Exponential stability and trajectory bounds of neural networks under structural variations.
IEEE Transactions on Circuits and Systems, **38**, 10 (Oct. 1991), 1182–1193.

[65] Wong, E. (1991)
Stochastic neural networks.
Algorithmica, **6**, 3 (1991), 466–478.

[66] Elgerd, O. I. (1967)
Control Systems Theory.
New York: McGraw-Hill, 1967.

[67] Kelley, H. J. (1962)
Method of gradients.
In G. Leitman (Ed.), *Optimization Techniques.*
New York: Academic Press, 1962.

[68] Gupta, N. K., and Mehra, R. K. (1974)
Computational aspects of maximum likelihood estimation and reduction in sensitivity function calculation.
IEEE Transactions on Automatic Control, **AC-19**, 6 (Dec. 1974), 774–783.

[69] Girosi, F., and Poggio, T. (1989)
Representations properties of networks: Kolmogororov's [1957] Theorem is irrelevant.
Neural Computation, **1**, 4 (Winter 1989), 465–469.

[70] Poggio, T., and Girosi, F. (1990)
Regularization algorithms for learning that are equivalent to multilayer networks.
Science, **247**, 4945 (Feb. 23, 1990), 978–982.

[71] Poggio, T., and Girosi, F. (1990)
Networks for approximation and learning.
Proceedings of the IEEE, **78**, 9 (Sept. 1990), 1481–1497.

[72] Girosi, F., and Poggio, T. (1990)
Networks and the best approximation property.
Biological Cybernetics, **63**, 3 (1990), 169–176.

[73] Frank, P. M. (1970)
Introduction to Sensitivity Theory.
New York: Academic Press, 1970.

[74] Nguyen, D. H., and Widrow, B. (1990)
Neural networks for self-learning control systems.
IEEE Control Systems Magazine, **10**, 3 (Apr. 1990), 18–23.

[75] Nguyen, D. H., and Widrow, B. (1989)
The truck backer-upper: An example of self-learning neural networks.
International Joint Conference on Neural Networks, Vol. 2, Washington, DC, June 18–22, 1989, 357–363.

[76] Widrow, B. (1990)
Adaptive inverse control.
In *Proceedings of SPIE–The International Society for Optical Engineering*, Vol. 1294, Orlando, FL, Apr. 8–20, 1990, 13–21.

[77] Widrow, B., and Lehr, M. A. (1990)
30 Years of adaptive neural networks: Perceptrons, madaline, and backpropagation.
Proceedings of IEEE, **78**, 9 (Sept. 1990), 1415–1442.

[78] Widrow, B., and Lehr, M. A. (1990)
Adaptive neural networks and their applications.
International Journal of Intelligent Systems, **8**, 4 (Apr. 1990), 453–507.

[79] Moutou, Y., Hamada, K., and Morishita, T. (1990)
Robust adaptive control of an inverted pendulum.
Transactions of the Society of Instrumentation and Control Engineers, **26**, 1 (Jan. 1990), 114–116.

[80] Meier Zu Farwig, H., and Unbehauen, H. (1990)
Discrete computer control of a triple-inverted pendulum.
Optimal Control Applications and Methods, **11**, 2 (Apr.–June 1990), 157–171.

[81] Anderson, M. J., and Grantham, W. J. (1990)
Lyapunov optimal control of a nonlinear inverted pendulum.
Transactions of American Society of Mechanical Engineers, **111**, 4 (Dec. 1990), 554–558.

[82] Hush, D. R., and Horne, B. G. (1993)
Progress in supervised neural networks: What's new since Lippmann?
IEEE Signal Processing Magazine, **10**, 1 (Jan. 1993), 8–39.

[83] Blum, A., and Rivest, R. L. (1988)
Training a 3-node neural network is NP-complete.
In *Proceedings of the Computational Learning Theory (COLT) Conference* (1988), 9–18.

[84] Kraft, L. G., An, E., and Ho, S. (1991)
Stability properties of CMAC neural networks.
Proceedings of American Control Conference, Vol. 2, Boston, MA, June 26–28, 1991, 1586–1591.

[85] Spall, J. C., and Cristion, J. A. (1991)
Efficient weight estimation in neural networks for adaptive control.
In *Proceedings of American Control Conference*, Vol. 1, Boston, MA, June 26–28, 1991, 16–20.

[86] Gschwendtner, A. B., and Widrow, B. (Eds.) (1988)
DARPA Neural Network Study.
MIT Lincoln Laboratory, Lexington, MA, 1988.

[87] Ng, K., and Lippmann, R. P. (1991)
A comparative study of the practical characteristics of neural network and conventional classifiers.
R. P. Lippmann, J. E. Moody, and D. S. Touretzky (Eds.), *Advances in Neural Information Processing Systems 3.*
San Mateo, CA: Morgan Kaufmann, 1991, 970–976.

[88] Godbole, D. N., and Sastry, S. S. (1993)
Approximate decoupling and asymptotic tracking for MIMO systems.
Proceedings of 32nd IEEE Conference on Decision and Control, Part 3 of 4.
San Antonio, TX, Dec. 15–17, 1993, 2754–2759.

[89] Burgio, G. D., Benedetto, M. D., and Sastry, S. S. (1992)
Adaptive linearization by dynamic state feedback: A case study.
Nonlinear Control System Design: Selected Papers from the 2nd IFAC Symposium.
Bordeaux, France, Jun. 24–26, 1992, 203–208.

[90] Ha, C. M. (1995)
Neural network approaches to AIAA aircraft control design challenge.
AIAA Journal of Guidance, Control, and Dynamics, **18**, 4 (July–Aug. 1995), 731–739.

[91] Sentoh, E., and Bryson, A. E. (1990)
A comparison of inverse control with optimal control.
AIAA Guidance, Navigation, and Control Conference: A Collection of Technical Papers, Part 2.
Portland, OR, Aug. 20–22, 1990, 1538–731–739.

[92] Cardell, N. S., Joerding, W., and Li, Y. (1994)
Why some feedforward networks can't learn some polynomials.
Neural Computation, **6**, 4 (June 1994), 761–766.

[93] Singh, S. N., and Yim, W. (1996)
Feedback linearization and solar pressure attitude control.
IEEE Transactions on Aerospace and Electronic Systems, **32**, 2 (Apr. 1996), 732–739.

[94] Slotine, J.-J. E., and Hendrick, J. K. (1993)
Robust input-output feedback linearization.
International Journal of Control, **57**, 5 (1993), 1133–1139.

[95] Jin, L., Nikiforuk, P. N., and Gupta, M. M. (1993)
Direct adaptive output tracking control using multilayered neural networks.
IEE Proceedings, Pt. D, **140**, 6 (Nov. 1993), 393–398.

[96] Warwick, K. (1994)
A critique of neural networks for discrete-time linear control.
International Journal of Control, **61**, 6 (1994), 1253–1264.

[97] Hunt, K. J., Sbarbaro, D., Zbikowski, R., and Gawthrop, P. J. (1992)
Neural networks for control systems—A survey.
Automatica, **28**, 6 (1992), 1083–1112.

[98] Zolghadri, A., Bergeon, B., and Monison, M. (1993)
A two ellipsoid overlap test for on-line failure detection.
Automatica, **29**, 6 (1993), 1517–1522.

[99] Zolghadri, A. (1993)
An algorithm for real-time failure detection in Kalman filters.
IEEE Transactions on Automatic Control, **41**, 10 (1993), 1537–1539.

[100] Kalman, R. E. (1965)
Irreducible realizations and the degree of a rational matrix.
Journal of Society for Industrial and Applied Mathematics (SIAM), **13**, 2 (June 1965), 520–544.

[101] Kalman, R. E. (1963)
Mathematical description of linear systems.
Journal of SIAM: Control, Series A, **1**, 2 (1963), 154–192.

[102] Kalman, R. E., Falb, P. L., and Arib, M. A. (1969)
Topics in Mathematical System Theory.
New York: McGraw-Hill, 1969.

[103] Antoulas, A. C. (Ed.) (1991)
Mathematical System Theory: The Influence of R. E. Kalman.
New York: Springer-Verlag, 1991.

[104] Brady, M. L., Raghavan, R., and Slawny, J. (1989)
Back propagation fails to separate where perceptrons succeed.
IEEE Transactions on Circuits and Systems, **36**, 5 (May 1989), 665–674.

[105] Cramer, H., and Leadbetter, M. R. (1967)
Stationary and Related Stochastic Processes.
New York: Wiley, 1967.

[106] Bucy, R. S., and Joseph, P. D. (1969)
Filtering for Stochastic Processes with Applications to Guidance.
New York: Wiley-Interscience, 1969.

[107] Calise, A. J. (1996)
Neural networks in nonlinear aircraft flight control.
IEEE AES Systems Magazine, **11**, 7 (July 1996), 5–10.

[108] Balakrishnan, S. N., and Biega, V. (1996)
Adaptive-critic-based neural networks for aircraft optimal control.
AIAA Journal of Guidance, Control, and Dynamics, **19**, 4 (July–Aug. 1996), 893–904.

[109] Napolitano, M. R., Chen, C. I., and Naylor, S. (1993)
Aircraft failure detection and identification using neural networks.
AIAA Journal of Guidance, Control, and Dynamics, **16**, 6 (1993), 999–1009.

[110] Napolitano, M. R., Neppach, C., Casdorph, V., Naylor, S., Innocenti, M., and Silvestri, G. (1995)
Neural-network-based scheme for sensor failure detection.
AIAA Journal of Guidance, Control, and Dynamics, **18**, 6 (Nov.–Dec. 1995), 1280–1286.

[111] Da, R. (1994)
Failure detection of dynamic systems.
AIAA Journal of Guidance, Control, and Dynamics, **17**, 2 (1994), 271–277.

[112] Da, R., and Lin, C.-F. (1996)
Sensitivity analysis algorithm for the state chi-square test.
AIAA Journal of Guidance, Control, and Dynamics, **19**, 1 (Jan.–Feb. 1996), 219–222.

[113] Wang, J., and Wu, G. (1996)
A multilayer recurrent neural network for on-line synthesis of minimum-norm linear feedback control systems via pole assignment.
Automatica, **32**, 3 (Mar. 1996), 435–442.

[114] Ha, C. M. (1995)
Neural networks approach to AIAA aircraft control design challenge.
AIAA Journal of Guidance, Control, and Dynamics, **18**, 4 (July–Aug. 1995), 731–739.

[115] Bakker, R., and Annaswamy, A. M. (1996)
Stability and robustness properties of a simple adaptive controller.
IEEE Transactions on Automatic Control, **41**, 9 (Sept. 1996), 1352–1357.

[116] Bakker, R., and Annaswamy, A. M. (1996)
Low order multivariable adaptive control with application to flexible structures.
Automatica, **32**, 3 (1996), 409–418.

[117] Klamka, J. (1991)
Controllability of Dynamical Systems.
Boston: Kluwer Academic Publishers, 1991.

[118] Johansen, T. A., and Foss, B. A. (1993)
Constructing NARMAX models using ARMAX models.
International Journal of Control, **58**, 5 (1993), 1125–1153.

[119] Brogan, W. L. (1989)
Algorithm for ranked assignments with applications to multiobject tracking.
AIAA Journal of Guidance, Control, and Dynamics, **12**, 3 (May–June 1989), 357–364.

[120] Mendel, J. M. (Ed.) (1994)
A Prelude to Neural Networks: Adaptive and Learning Systems.
Englewood Cliffs, NJ: Prentice-Hall, 1994.

[121] Kerr, T. H. (1997)
A critical perspective on some aspects of GPS development and use.
Proceedings of 16th Digital Avionics System Conference, Irvine, CA, Oct. 25–30, 1997.

[122] Ho, Y.-C. (1997)
Book review of "Neuro-Fuzzy and Soft Computing—A Computational Approach to Learning and Machine Intelligence."
By J. S. R. Jang, C. T. Sun, and E. Mizutani (Prentice-Hall, 1997), *IEEE Transactions on Automatic Control*, **42**, 10 (Oct. 1997), 1482–1484.

Thomas H. Kerr received the B.S.E.E. (magna cum laude) from Howard University, Washington, DC, in 1967 and the M.S. and Ph.D. degrees in electrical engineering from the University of Iowa, Ames, in 1969 and 1971, respectively, in control and estimation.

From 1971–1973, he worked as a Controls Engineer at G.E.'s Corporate Research & Development Center in Schenectady, NY. From 1973–1979, he was a member of the Technical Staff at TASC, developing real-time failure detection algorithms for SSBN submarine SINS/ESGN navigation and evaluating/improving algorithms and strategies for SSBN navaid fix taking. From 1979–1986, he was a Senior Analyst/Systems Engineer at Intermetrics, Inc., involved in applying and evolving decentralized Kalman filters for JTIDS RelNav refinement & ICNIA decentralized filter development. From 1986–1992, he was a member of the staff at Lincoln Laboratory of MIT, involved in radar target tracking of reentry vehicles (RV's), developing credible RV decoys, and in performing GPS navigation analysis. He is currently CEO and one of the Principal Investigators at TeK Associates, an engineering consulting, R&D, and software development company that he founded in 1992. He has an ongoing consulting affiliation with MITRE in National Missile Defense and has implemented a Cramer–Rao lower bound evaluation methodology for gauging target tracker performance.

Dr. Kerr is the developer and author of the commercially available menu-driven PC software for Kalman filter/smoother/Monte-Carlo simulation and/or LQG regulator control analysis/implementation with on-line tutorials on its use. He is a member of IEEE AC, AES, SP, IT, and several other IEEE Societies, AIAA GC&D, ION, SPIE, MAA, ASA, AAAS, SBANE, CRW, and past chairman (1990–1992) and vice-chairman (1994–1996) of the local Boston section of the IEEE Control System Society.