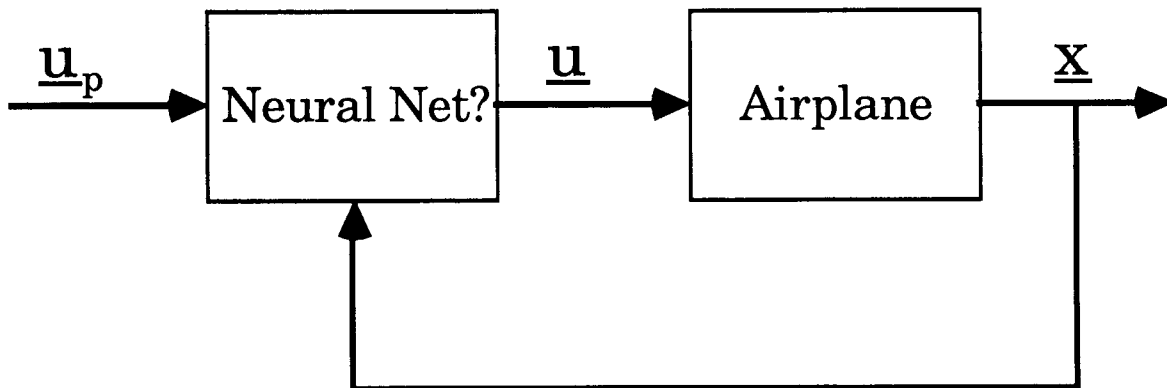# Neural Networks for Aircraft Control

Dennis Linse
Department of Mechanical and Aerospace Engineering
Princeton University

Current research in Artificial Neural Networks indicates that networks offer some potential advantages in adaptation and fault tolerance. This research is directed at determining the possible applicability of neural networks to aircraft control. The first application will be to aircraft trim.

$\underline{u}_p \longrightarrow$ Neural Net? $\xrightarrow{\underline{u}}$ Airplane $\xrightarrow{\underline{x}}$

- Introduction to Neural Networks

- Neural Networks for Aircraft Trim

- Application of Neural Networks to Control

Introduction to Neural Networks

Artificial neural networks (usually called simply neural networks) are an attempt to model the processing behavior of the nervous system. Neural network research focuses on a much larger group of networks loosely classified by the dense interconnection of simple computational elements. Since the majority of the processing operations are independent of one another, neural networks can conduct massively parallel computations. While the human brain is estimated to have 100 billion neurons each with approximately 1000 inputs and outputs, artificial neural networks are interested in the computational capabilities of smaller networks which may or may not be biologically correct.

- Artificial neural networks

> Abstract simulation of real nervous system

> Dense interconnection
>> of simple computational elements

> Massively parallel computation

> Biologically inspired vs. biologically accurate

> Estimated 100 billion (10^{11}) neurons in brain
>> Each has 1000 inputs and outputs

Introduction to Neural Networks

Most current applications of neural networks have been in the area of pattern recognition. A smaller number of applications have been devoted to optimization, most notably the famous Traveling Salesman Problem in which the shortest route through selected cities is determined.

The major benefits of neural network are two-fold. First, neural networks have the ability to learn their internal knowledge from presentation of input-output data. This means that the network does not have to be programmed in the traditional sense. It must be trained with examples of the desired input-output relationship. Adaptability arises if the learning process continues while the network is in operation. Neural networks are potentially very fault tolerant due to the massively parallel architecture. The knowledge contained in the network is distributed throughout the network so that the loss of individual computational elements should not seriously degrade the performance of the network.

- Applications

    Speech/Image/Pattern Recognition, Classification and Restoration

    Optimization

- A few potential benefits

    Adaptability/Learning

    Robustness / Fault tolerance

Introduction to Neural Networks

Neural network models are differentiated by the type of computational element (or node), the connection of the nodes, and the learning rule used to update the weighted connections between nodes.

- Neural network models are specified by

  Node characteristics

  - linear / nonlinear
  - analog / discrete (binary)

  Net topology

  - node interconnections

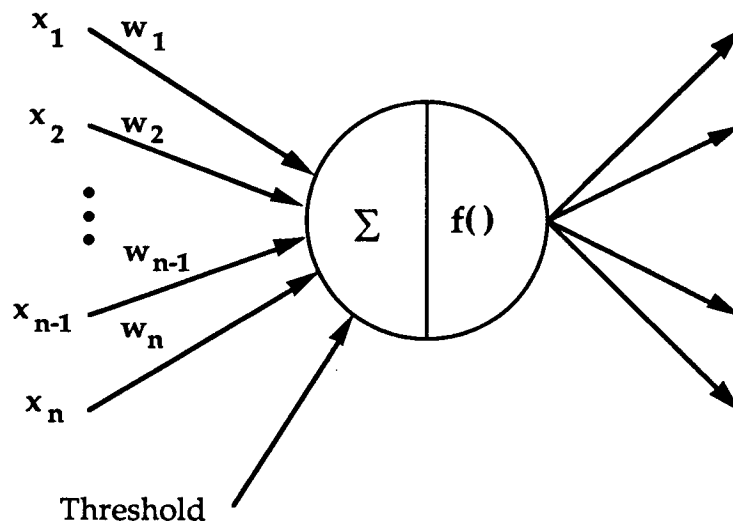   - layers
    - feedforward/feedback

  Learning rule

  - supervised / unsupervised

Neural Network Node Characteristics

An individual neural network node computes a weighted sum of its inputs and runs the sum through a fixed function. The resulting value is transmitted to all of the following network elements. The fixed function, called the node activation function, is generally a bounded, S-shaped non-linear element. A common activation function is the sigmoid function. The nonlinearity of the activation function provides the computational power of the network. Networks with linear elements have been tested, but most contain some type of nonlinear function.

- Individual Neural Net Node

Threshold

- Node Activation Function

Sigmoid $f(x) = 1.0 / (1.0 + \exp(-x))$
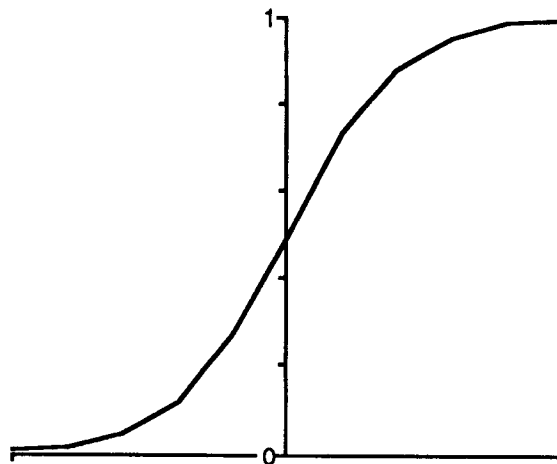
Network Topology and Operation

The multilayer feedforward network is characterized by the distinct layers of network nodes connected only in the forward direction. There are no connections between nodes in a single layer nor are there connections from higher layers to lower layers. A constant unit input is used as a threshold or bias.

*The operation of the network is easily characterized by a recursive matrix equation. The output of each layer is simply the weighted sum of its inputs passed through the nonlinear activation function f. The knowledge of the system is contained in the weighted connections, i.e., the weights, **W**. As the weights change, the nonlinear input-output relationship function modeled by the network changes.*

- Multilayer Feedforward Network
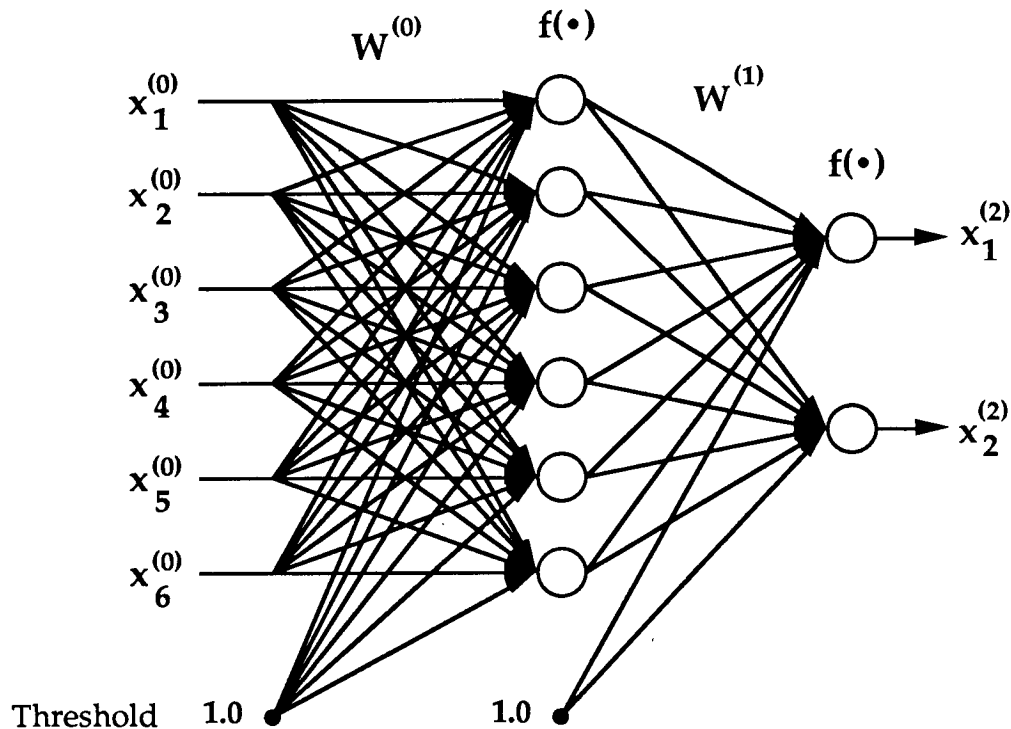
- Forward Operation (Activation, Retrieval)

$$\underline{x}^{(k)} = \underline{f}(\ \mathbf{W}^{(k-1)}\ \underline{x}^{(k-1)}\ ), \qquad k = 1, \dots, N \text{ layers}$$
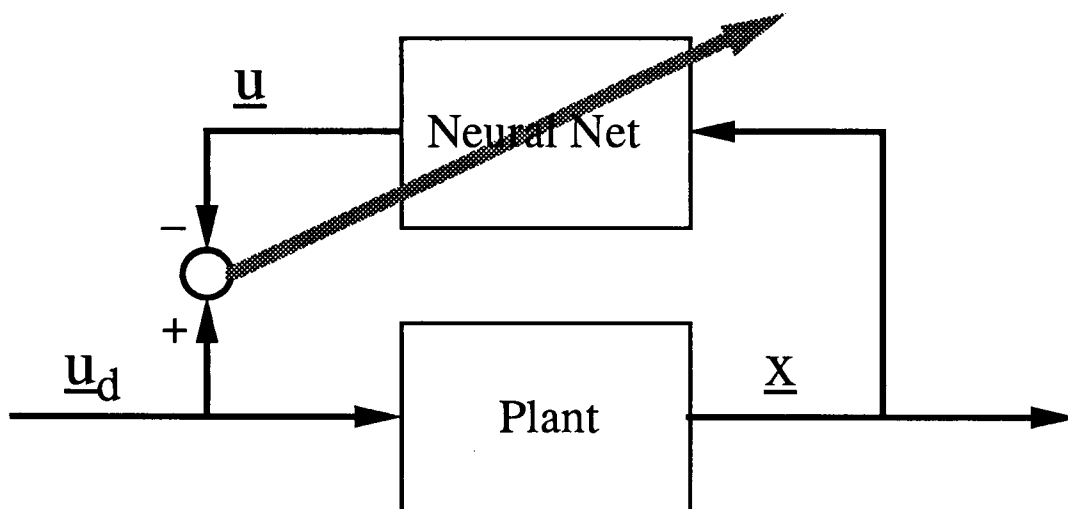
$$\underline{x}^{(0)} = \text{input}$$

$$\underline{x}^{(N)} = \text{output}$$

Neural Network Learning

In supervised learning, the neural network is trained to match a particular input-output relationship. First an input is applied to the neural network. Next, the output produced by the network is compared to the desired output. Finally, the network weights are adjusted based upon the error.

- Supervised Learning Diagram

Neural Network Learning

The backpropagation learning algorithm adjusts the weights of the network to minimize the mean square error of all of the outputs. The difficulty comes when assessing the importance of individual weights of the internal (or hidden) layers on each output. The backpropagation learning algorithm explicitly assigns a portion of the error to each element. The one requirement is that the nonlinear function in each network node must be differentiable.

- Learning by Backpropagation

    Adjust weights to minimize the mean square error

    Difficulty of error assignment

    Differentiability of nonlinearity

- Backpropagation

    $$W_{i+1} = W_i + \Delta W_i$$

    $$\Delta W_i = \beta \, \underline{x}^{(k-1)} \, \underline{\delta}^{(k)}$$

    $\underline{\delta}$ is the error term

    for output layer $= (\underline{u}_d - \underline{u})$

    for hidden layers, error is backpropagated

Neural Network for Aircraft Trim

A multilayer feedforward neural network is trained on the input-output relationship for the longitudinal trim of a transport aircraft. Using the backpropagation learning rule, the network is taught to produce the aircraft control positions necessary to maintain a given trim state. Such a trim map could be used as part of a perturbation control law. The power of the neural network would eventually be in the ability to continuously update its knowledge of the aircraft during operation.

- Exploit learning/adaptability features of neural networks

- Teach neural network the map between
  trim states and trim control positions

  Useful for

  > Autotrim system
  > Trim map for perturbation control law

- Straight and level flight
  Longitudinal states and controls for learning

  $\underline{x} = \{$ $\bar{q}$ - dynamic pressure

  V - forward velocity

  γ - flight path angle
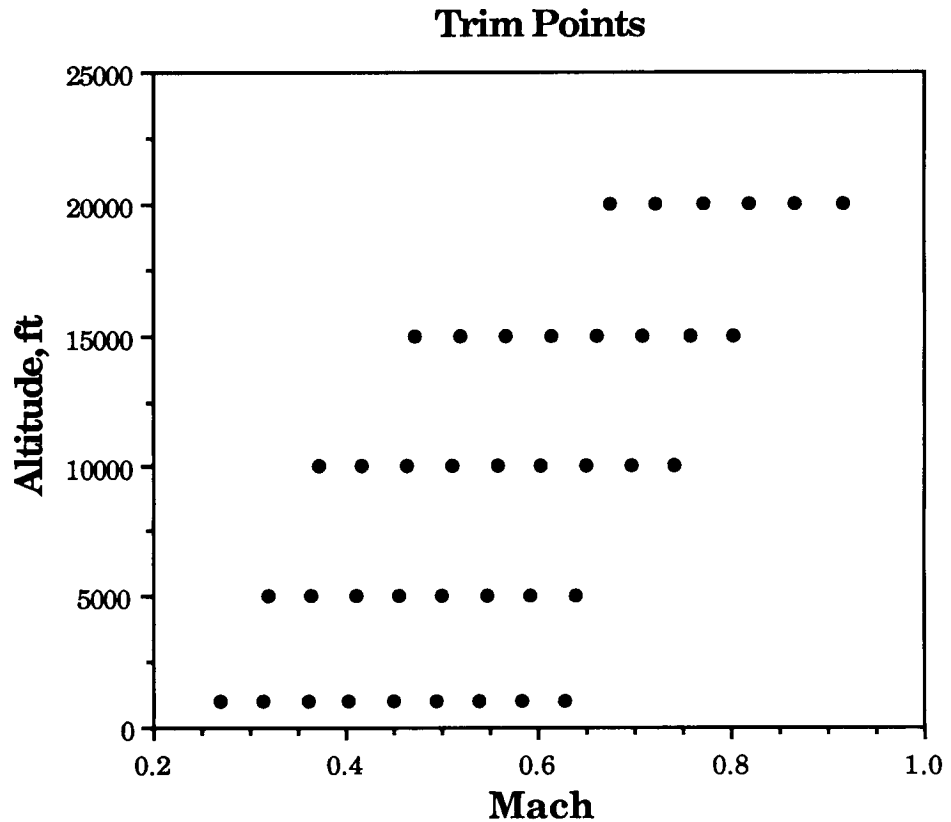
  α - angle of attack

  q - pitch rate

  h - altitude $\}$

  $\underline{u} = \{$ δ_E - elevator deflection

  δ_T - throttle deflection $\}$

Aircraft Trim Points

The neural network is trained on aircraft trim points ranging from Mach 0.25 at 1000 feet up to Mach 0.9 at 20000 feet.

Trim Points

Example Network Run

A simple example is shown in which the network is trained only on the aircraft data at 1000 and 5000 foot altitudes. A three layer network is used with 6 inputs, 6 hidden nodes in one layer, and 2 outputs.

- Train network with 17 trim points, 6 units in hidden layer

- 5000 Iterations, $\beta = 0.5$, momentum = 0.9

- Final Total Square Error = 0.0004

Error in Learned Response

The results after 5000 iterations are quite good. The elevator error is less than 1 percent and the throttle error is less than 3 percent over the entire learned range.
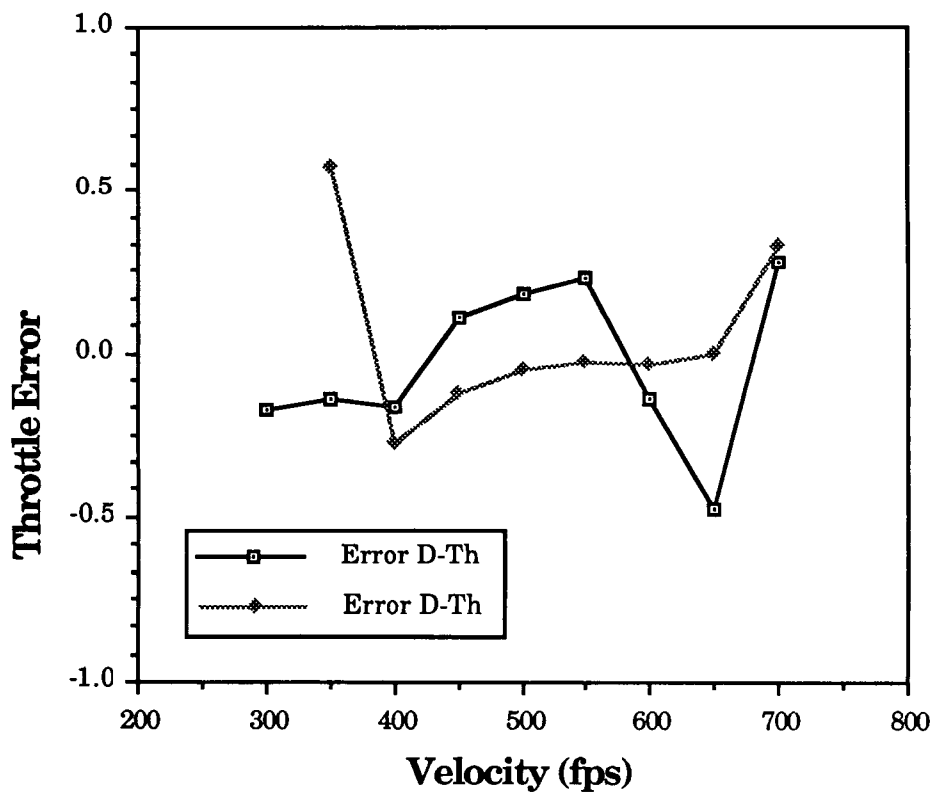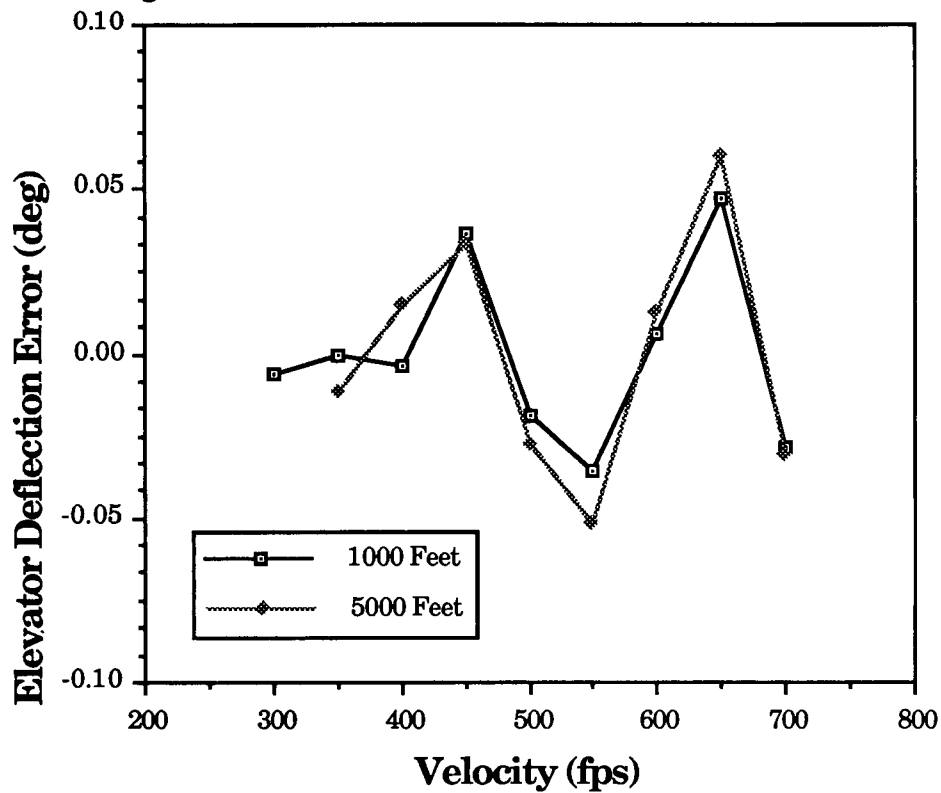
# Neighboring Optimal Control Law
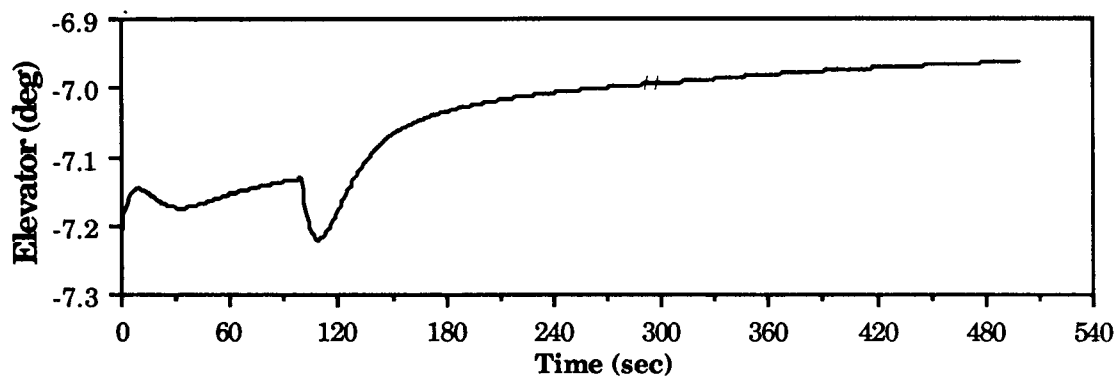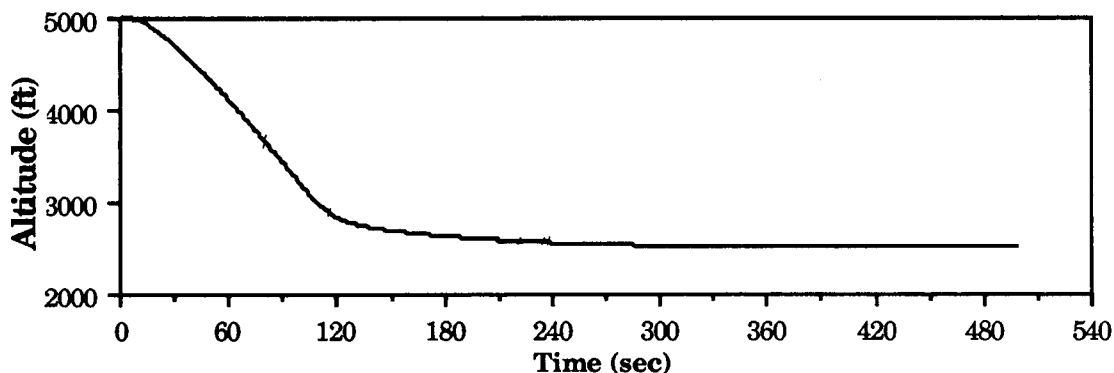# using Neural Network

The neural network trained as a trim map fits quite nicely into a neighboring optimal control scheme. A perturbation control law is used to eliminate the errors between the actual aircraft state and the desired trim value. The resulting perturbation control position is combined with the trim control position from the neural network trim map to produce the control input for the aircraft.

- Full state feedback perturbation control law developed for linearized system

- Applied to full nonlinear system using a Neighboring Optimal Control structure

- Trim values provided by Neural Network
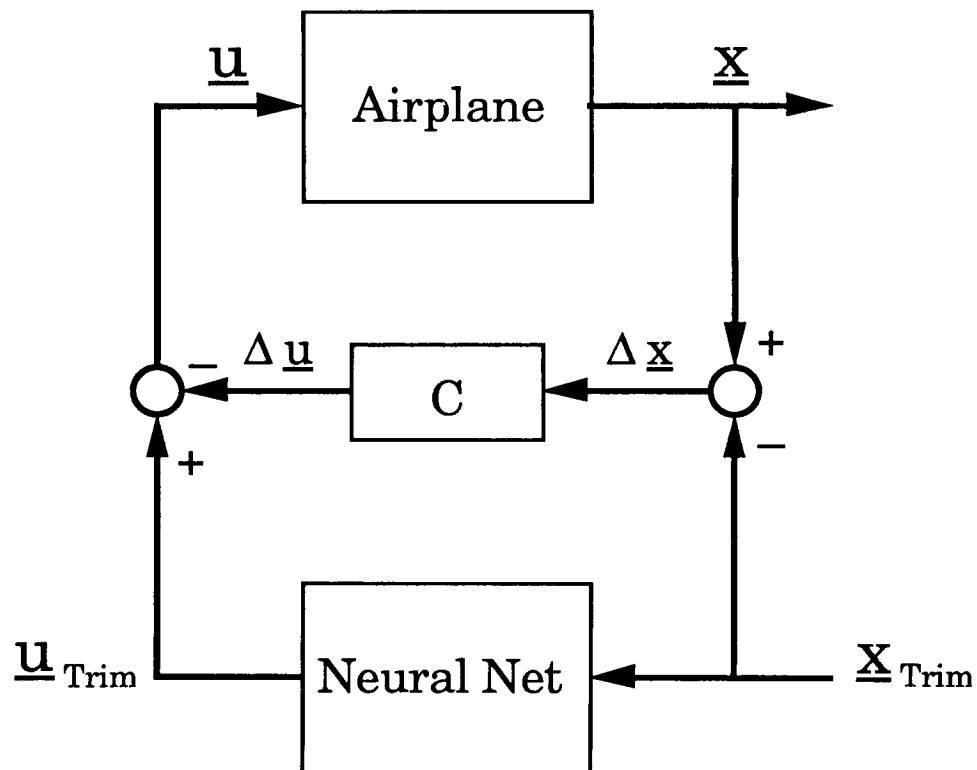
# Example Histories using Neighboring Optimal Control with a Neural Net

Using the neighboring optimal control scheme with a neural network as the trim map, a smooth transition from 5000 to 2500 feet is effected for the simulated aircraft. The control positions are smooth and accurate over the entire range.
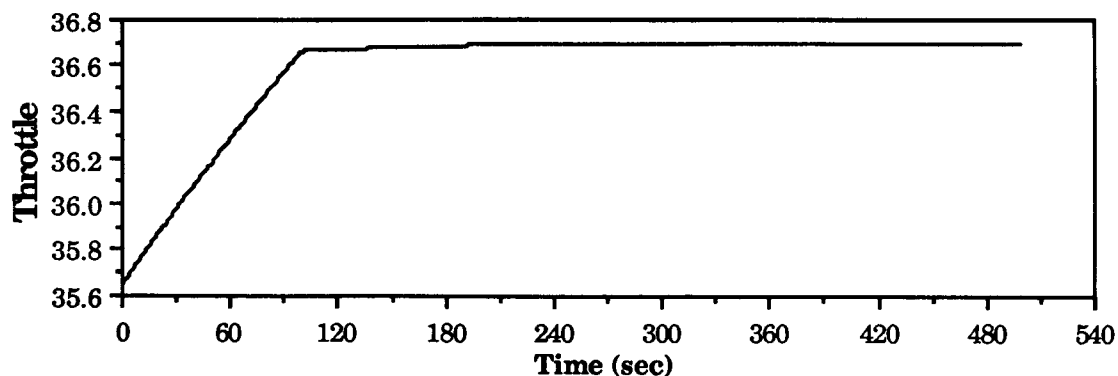
Neural Networks for Control

In addition to use as a trim map, a neural network could possibly be trained as a feedback control element. If the aircraft is modeled by a discrete time linear system, the weights could be adjusted as to minimize the error between the actual system states and some desired states. The mathematics of such a minimization appears feasible.

- Investigate Neural Network as Feedback Control Element
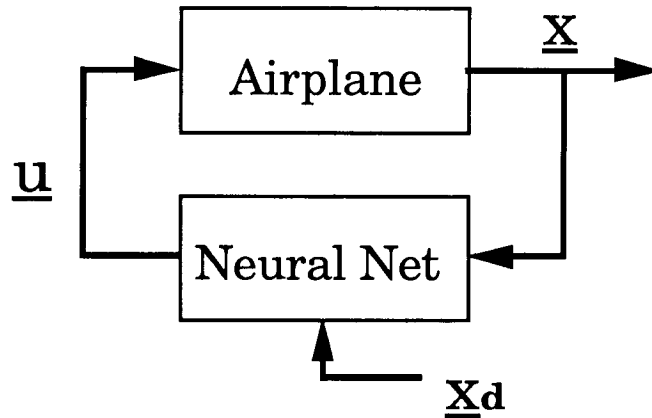
- Discrete Time Linear System

$$\underline{x}_{i+1} = \Phi \, \underline{x}_i + \Gamma \, \underline{u}_i$$

- Neural Network Control Law

$$\underline{u}_i = \underline{f}(\, W^{(1)} \, \underline{f}(\, W^{(0)} \, \underline{x}_i) \,)$$

- Adjust weights to minimize least squares fit error

$$I = \frac{1}{2} \sum_{i=1}^{M} (\underline{x}_d(i) - \underline{x}(i))^T (\underline{x}_d(i) - \underline{x}(i))$$

- Steepest descent training based on error gradient

$$W^{(1)}_{ij_{k+1}} = W^{(1)}_{ij_k} - \beta \, \frac{\partial I}{\partial W^{(1)}_{ij_k}}$$

Neural Networks for Aircraft Control

There are many topics which still must be investigated before any valid conclusions may be made about the usefulness of neural networks in aircraft control. The ability of the network to correctly generalize to trim points between the trained data is an important topic which must be looked into. The fault tolerance of neural networks is based upon massive parallelism. It is unknown how much of this fault tolerance is retained by smaller networks. There are many more possible applications for neural networks that are, as yet, untapped.

Topics of Interest for Neural Networks

- Generalization capabilities of networks

    What happens between learned points?

- Fault tolerance capabilities of networks

    What happens if nodes or weights malfunction?

- Other uses of neural networks for aircraft control

    Model identifier

    Adaptive control element

Conclusions

- Neural Networks show some promise for control applications

- There is much left to investigate