Verification and Validation of Adaptive and Intelligent Systems with Flight Test Results

John Burken & Dick Larson

NASA, Dryden Flight Research Center

Edwards CA USA

UCAUV 2009

3 &4 April 2009

Bangalore, India

Outline

- Background Information
  - Problem statement
- Project Goals / Objectives
  - Motivation
- System Overview Architecture
- Adaptive Control System Design
  - Neural Network Approach
- Flight Approach
- V&V Task
  - Loads 80%
  - G load limits Pilot limits +/- (self imposed)
- Flight Results

Background Information

Problem Statement

- **How do you V&V a Piloted Adaptive system?**
- **Constraints:**
  - **Piloted aircraft (Modified F-15)**
    - **Pilot limited any transients to +/-.5 lateral gees & +/-2 longitudinal.**
  - **Flight Control Computers (quad system) (Level A)**
  - **The adaptive algorithms are processed on a single string system called ARTSII (Level B)**
  - **Any maneuver can not exceed 80% structural load limit.**

F-15 IFCS Project Goals

- Demonstrate Control Approaches that can Efficiently Optimize Aircraft Performance in both Normal and Failure Conditions [A] & [B] failures.

- Advance Neural Network-Based Flight Control Technology for New Aerospace Systems Designs with a Pilot in the Loop

Gen II Objectives

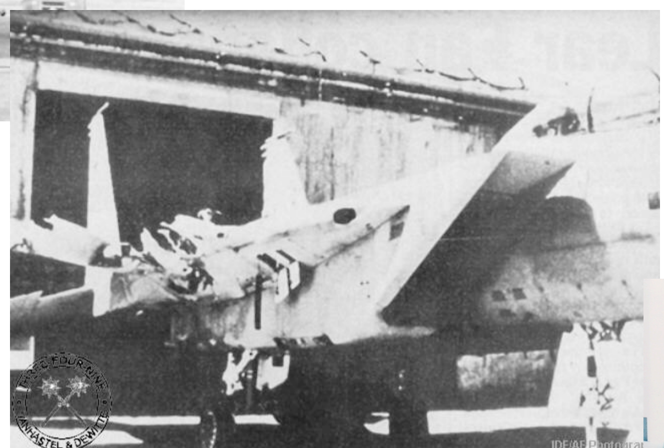
- Implement and Fly a Direct Adaptive Neural Network Based Flight Controller

- Demonstrate the Ability of the System to Adapt to Simulated System Failures
  - Suppress Transients Associated with Failure
  - Re-Establish Sufficient Control and Handling of Vehicle for Safe Recovery

- Provide Flight Experience for Development of Verification and Validation Processes for Flight Critical Neural Network Software

Motivation

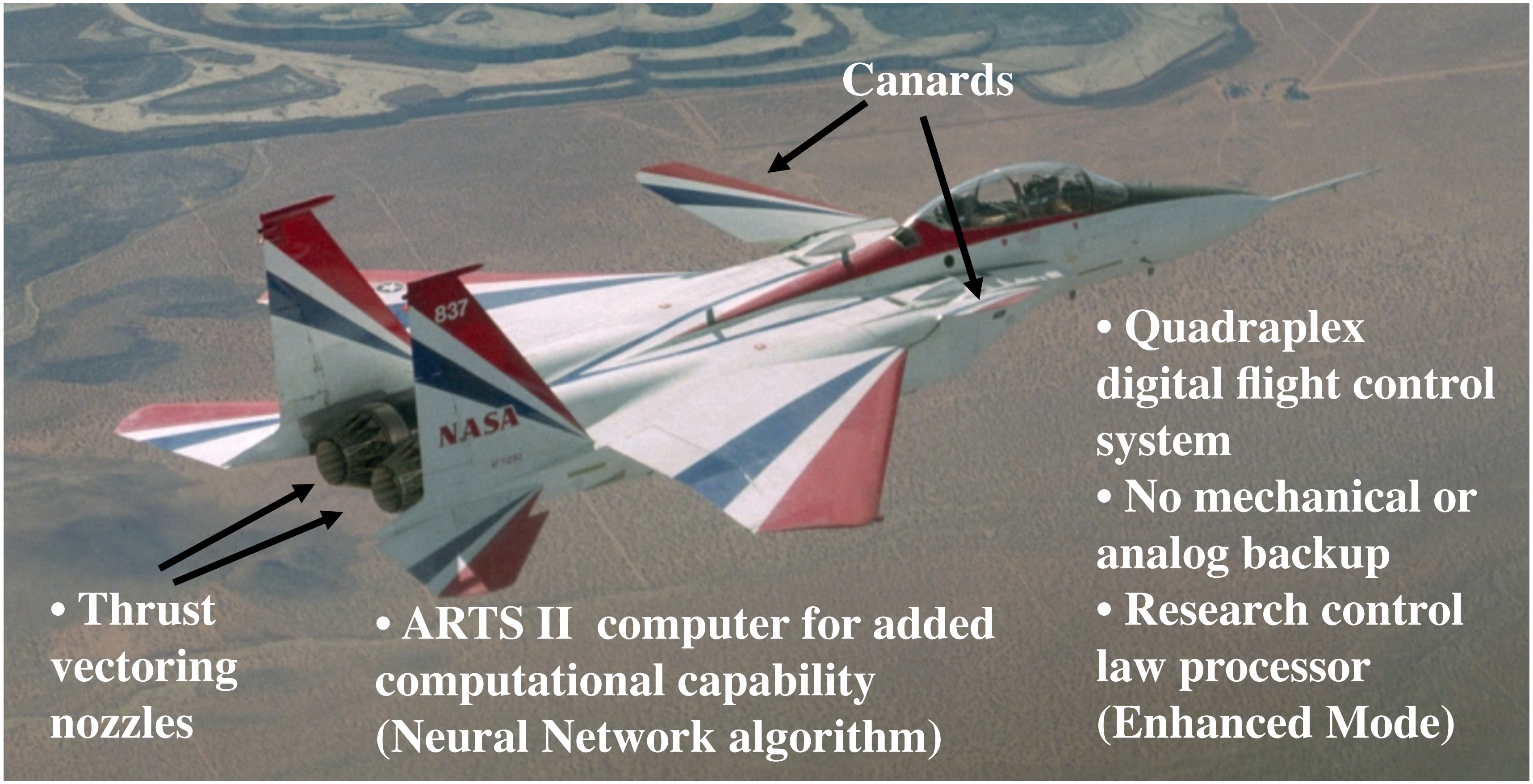
These are survivable accidents



IFCS has potential to reduce the amount of skill and luck required for survival

NASA NF-15B Tail Number 837

Extensively modified F-15 airframe



V & V Issues

- **System Overview Architecture**
  Quad digital flight control computers (Level A)

- **Airborne Research Test System II (ARTSII)** (single string Level B)

- **Project tried to prove stability issues using Lyapunov methods for V&V but was not conclusive.**

- **Assumptions:**
  - **Single string signals may go hard over any time.**
  - **Note: If you had a dual redundant ARTSII the V&V task would be different then this projects V&V.**

V & V Issues

- **System Overview Architecture**
  Quad digital flight control computers (Level A)

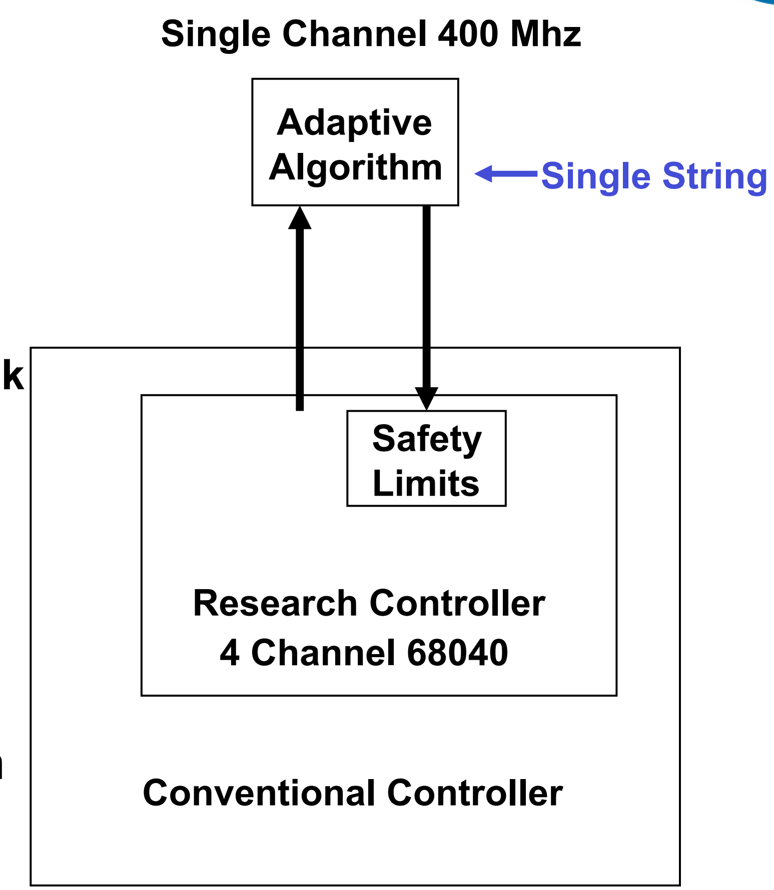
- **Airborne Research Test System II (ARTSII)** (single string Level B)

- **Project tried to prove stability issues using Lyapunov methods for V&V but was not conclusive.**

- **Assumptions:**
  - **Single string signals may go hard over any time.**

Limited Authority System

- Adaptation algorithm implemented in separate processor
  - Class B software
  - Autocoded directly from Simulink block diagram
  - Many configurable settings
    - Learning rates
    - Weight limits
    - Thresholds, etc.
- Control laws programmed in Class A, quad-redundant system
- Protection provided by floating limiter on adaptation signals

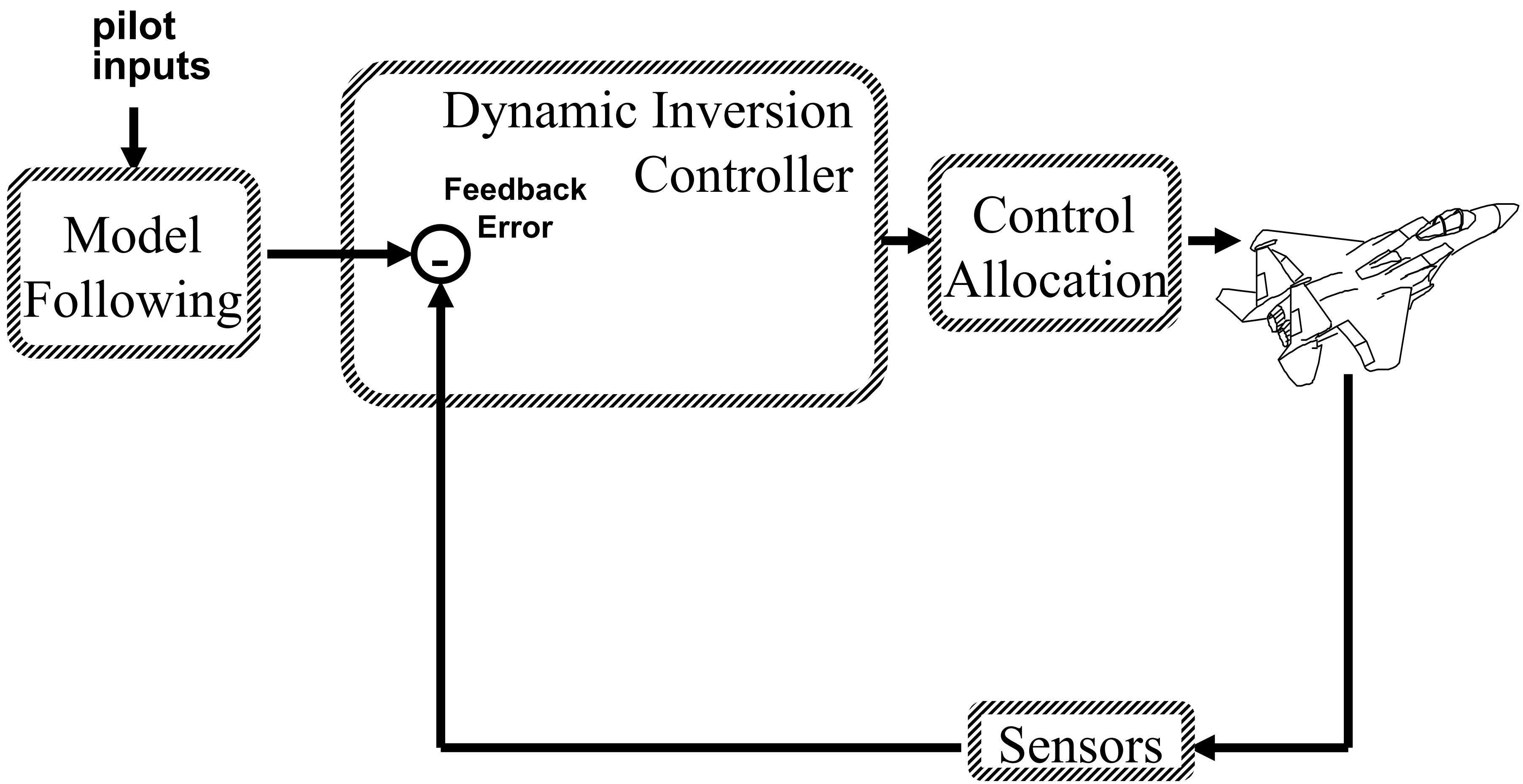


ARTS II

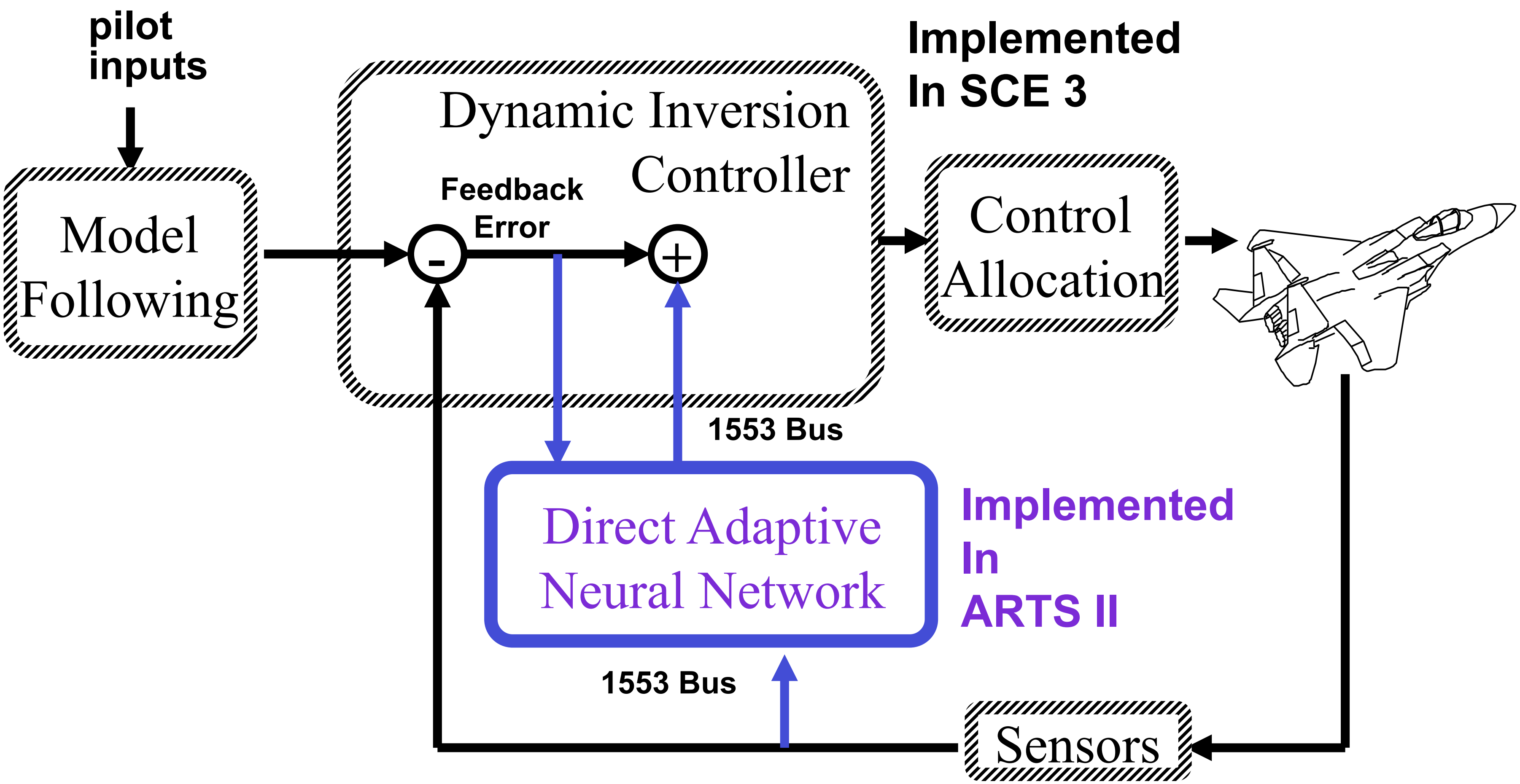
- **Provides Added Capacity (Throughput and Memory) to Run the IFCS Advanced Algorithms**

- **Airborne Research Test System II (ARTS II)**
  - VME Based System
  - 3 Single Board Computer (SBC) Processor Cards
    - 1553 Interface
  - PowerPC 750
    - 400MHz Operating System
    - 66Mhz Local Bus
  - 1 MB L2/Cache
  - 128 MB of SDRAM

Gen II Architecture (non Adaptive)

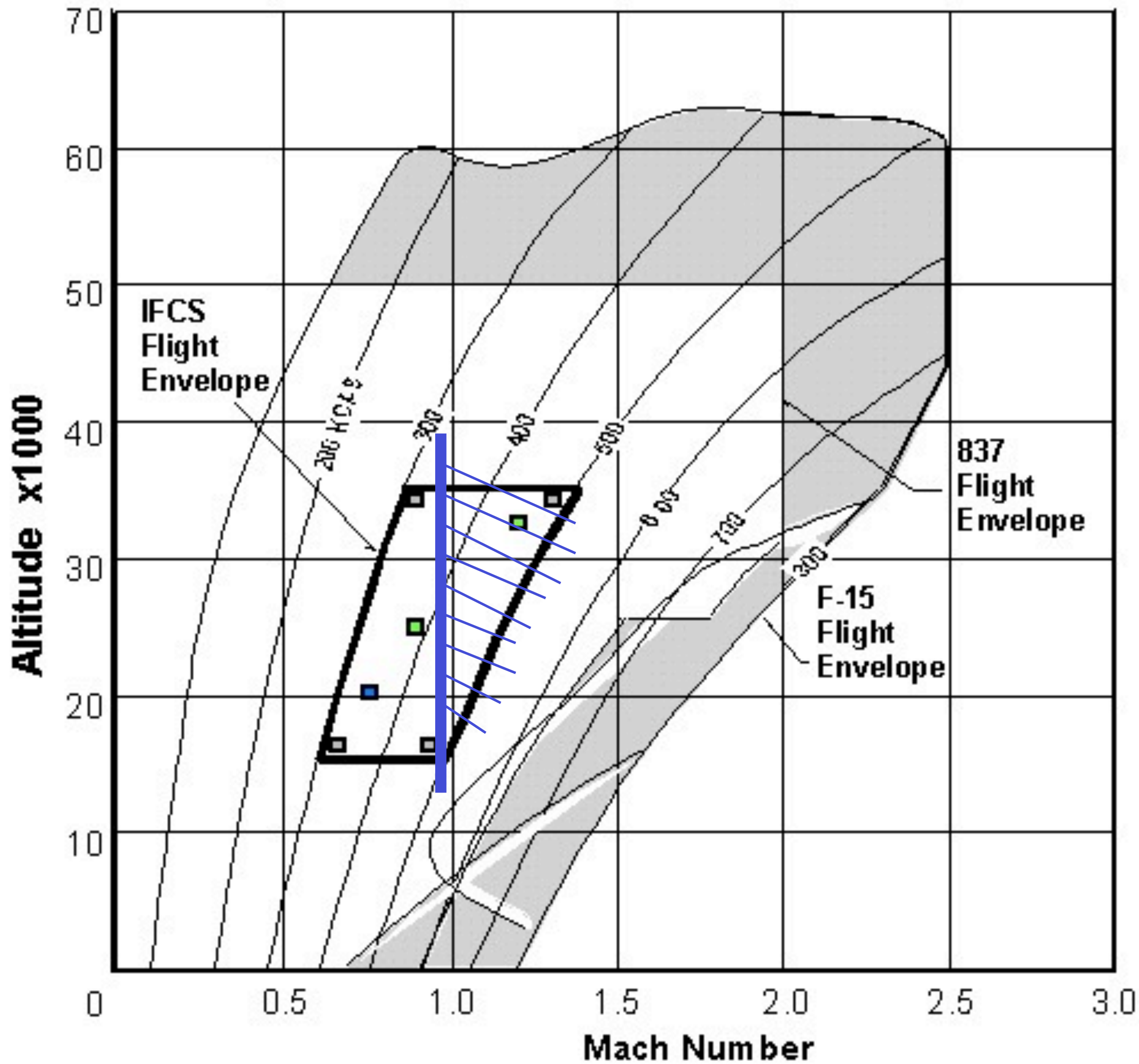


Gen II Architecture
(Adaptive)



Flight Envelope

For Gen 2
Mach < 0.95



Flight Experiment

- Assess Handling Qualities of Gen II Controller without Adaptation
- Activate Adaptation and Assess Changes in Handling Qualities
- Introduce Simulated Failures
  - Control Surface Locked (“B Matrix Failure”)
  - Angle of Attack to Canard Feedback Gain Change (“A Matrix Failure”)
- Report on “Real World” Experience with a Neural Network Based Flight Control System

Direct Adaptive Neural Network

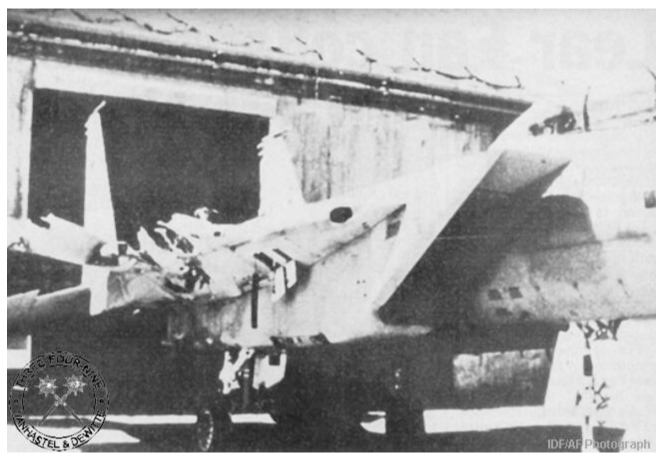
Neural Network Design and Implementation for the F-15 837

April, 2009

F-15 Intelligent Flight Control Systems

- Motivation / Problem Statement {The Big Picture}
  - Land a damaged airplane or, return to a safe ejection site.

- General Goals & Objectives
  - Flight evaluation of neural net software.
  - Increased survivability in the presence of failures or aircraft damage.
    - Increase your boundary of a flyable airplane.
    - Increase your chances to see another day.
    - Increase your chances to continue the mission.



Background : Historical Note

- **Neural Networks are a subset of Adaptive Control.**
- **Adaptive Control Research Started in the early 1950’s.**
  - **Auto-Pilot work (non-Neural Network).**
- **Research Diminished due to the crash of X-15.**
  - **Reference: Eugene Lavretsky, “Adaptive Control: Introduction, Overview, and Applications.”**

General Neural Network Problem Statements Plus Others

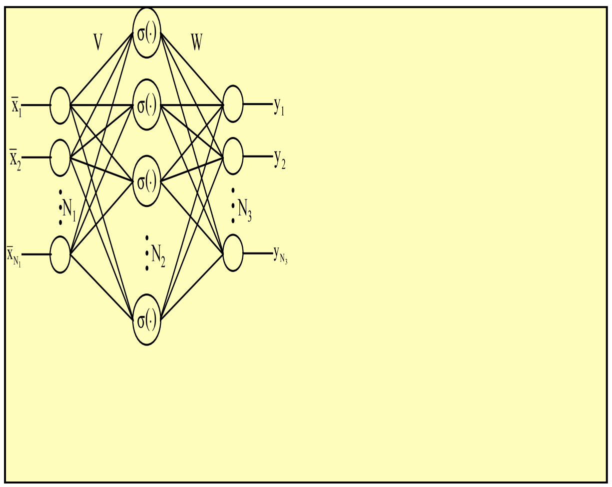
- **Why Use a Neural Network?**
- **How much do Neural Networks help a controller?**
- **Why Use Dynamic Inverse Control?**
- **How much do Neural Networks cost w.r.t. compute power?**
- **How can we certify a Neural Network?**
- **Some of these questions are NOT answered in this presentation**

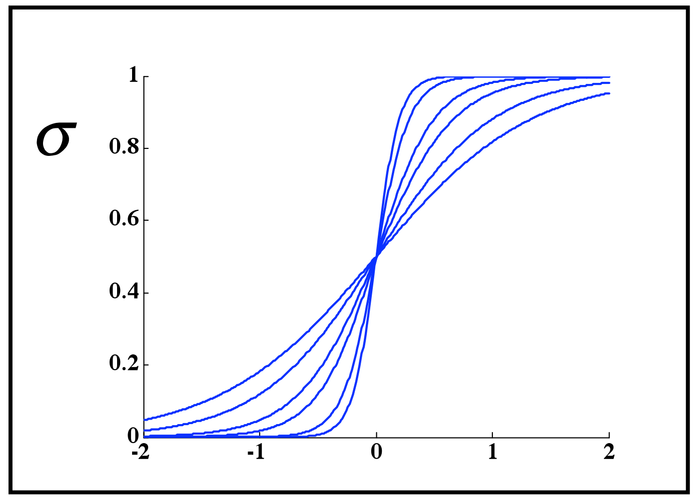
Why Neural Networks?

Neural Networks are Universal Approximators

Minimizes a H^2 norm

They permit a nonlinear parameterization of uncertainty



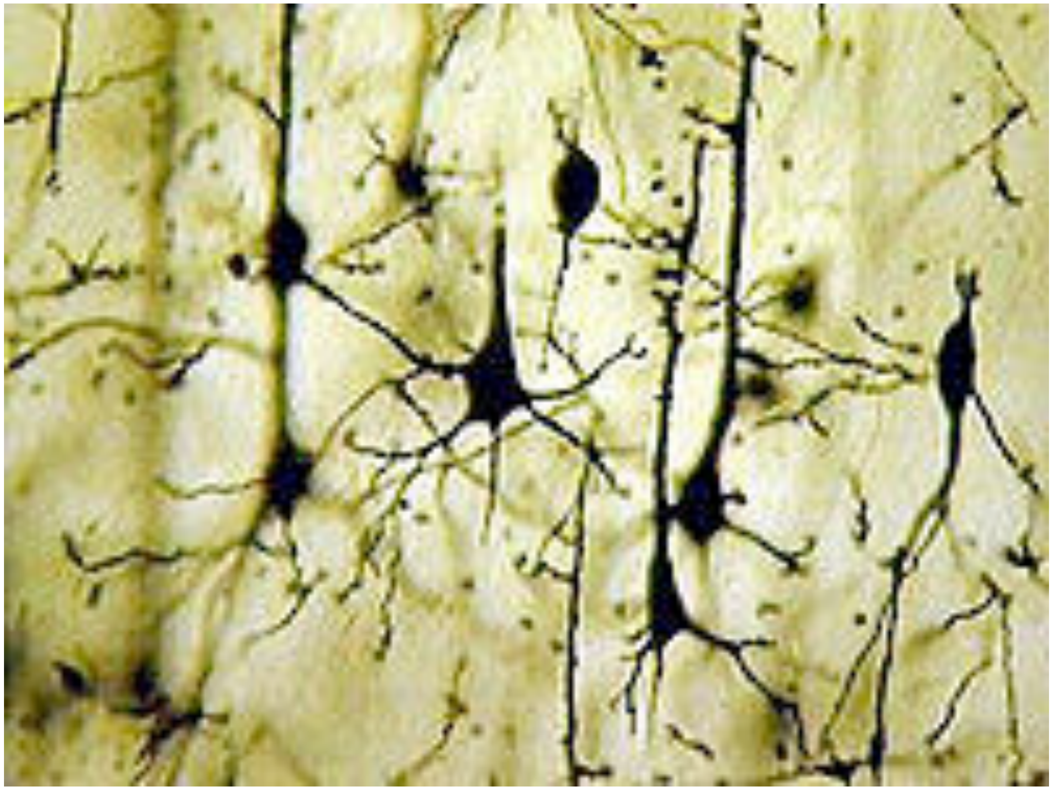


$$y = f(x) = W\sigma(Vx) + \varepsilon(x)$$

$$|\varepsilon(x)| < \varepsilon^* \ \ \forall \ x \,\varepsilon\, \Omega$$

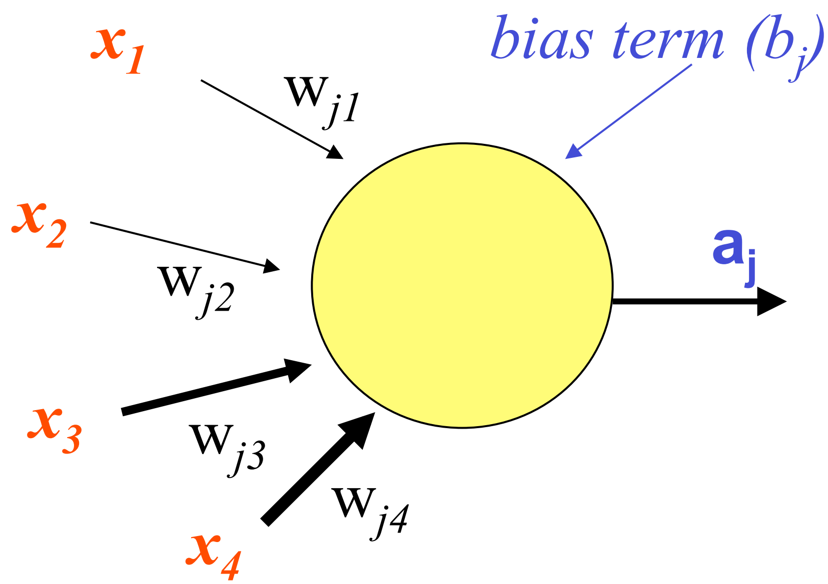
$$\dot{W} = -\left[\left(\sigma - \sigma' V^T \bar{x}\right)\eta + \kappa \|e\| W\right]\Gamma_W$$

Neurons in the human brain



Neural networks simulate the activity of biological neurons within the human body. Neural networks are implemented in an attempt to re-create the learning processes of the brain by recognizing patterns.

Single Neuron

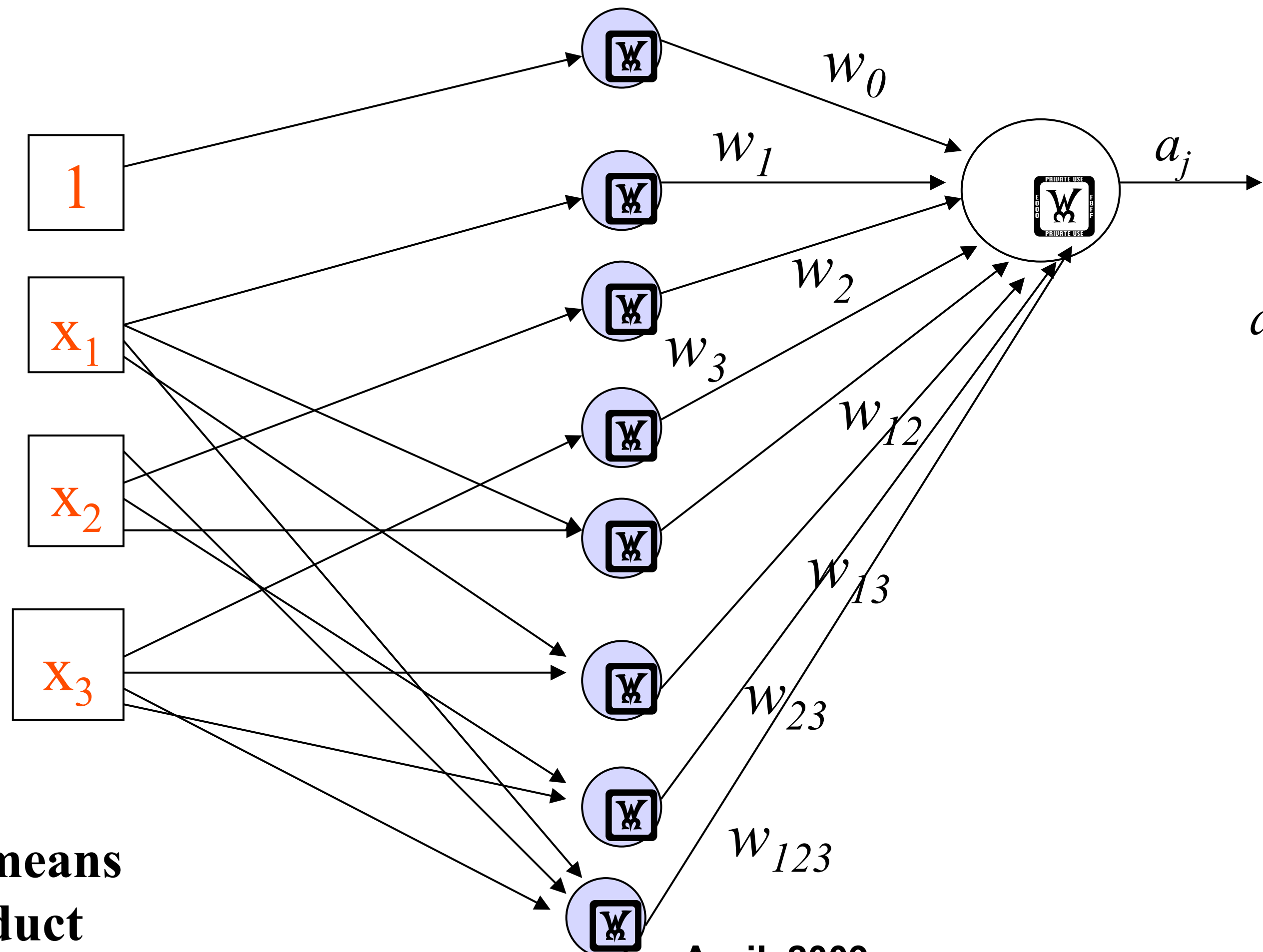


Combination function, $a_j = \sum w_{jk} * x_k + b_j$

summation   product

Multiple neurons

For 1 neuron with 3 inputs:



$$
\begin{aligned}
a_j = \ & w_0 1 + \\
& w_1 x_1 + \\
& w_2 x_2 + \\
& w_3\ x_3 + \\
& w_{12}\ x_1 x_2 x + \\
& w_{13}\ {}_1 x_3 x_2 + \\
& w_{23}\ x_3 + \\
& w_{123} x_1 x_2 x_3
\end{aligned}
$$

⊠ means product

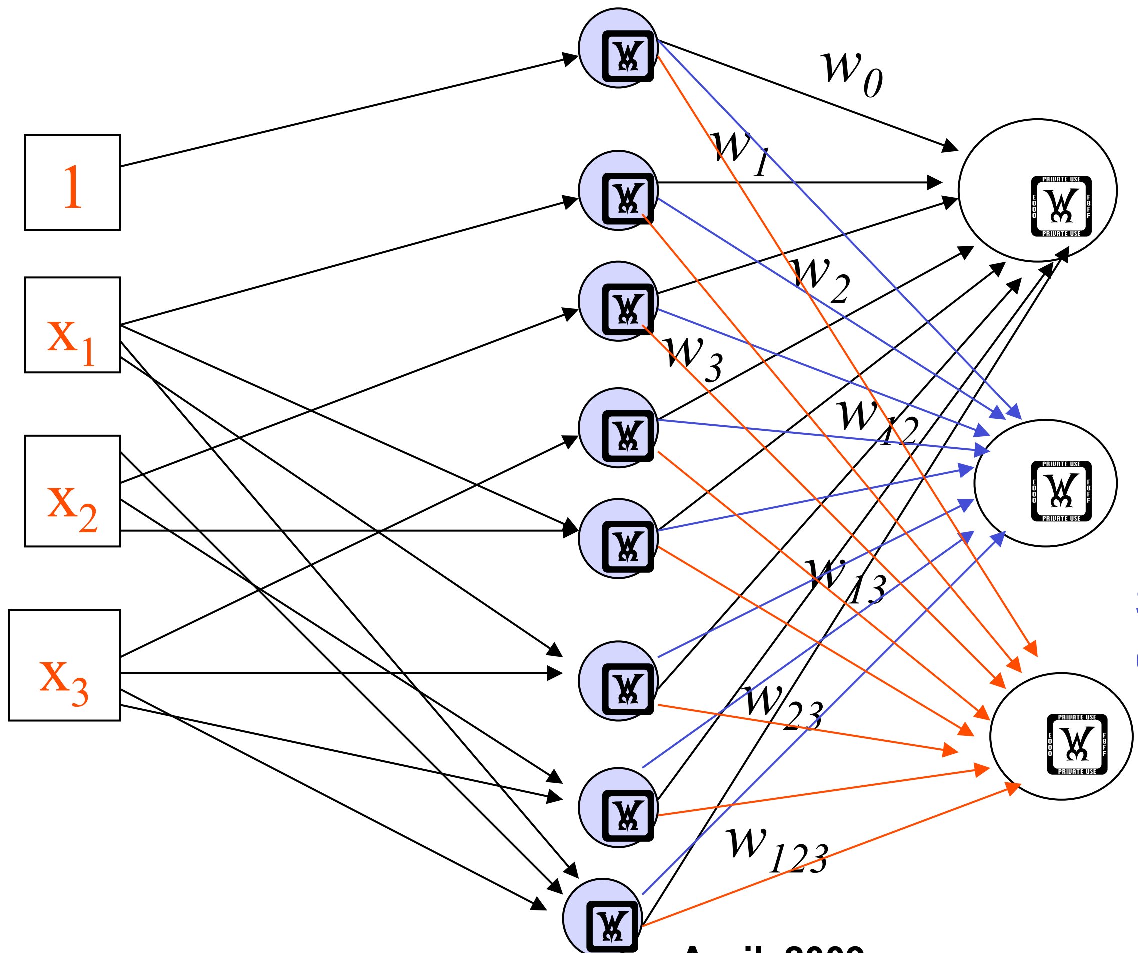
Activation Function for fully connected neuron

- **Activation function for one neuron is written mathematically in a general form as:**

$$a_j = w_j^{(0)} + \sum_{i_1=1}^{d} w_{j i_1}^{(1)} x_{i_1} + \sum_{i_1=1}^{d} \sum_{i_2=1}^{d} w_{j i_1 i_2}^{(2)} x_{i_1} x_{i_2} + \underbrace{\sum_{i_1=1}^{d} \sum_{i_2=1}^{d} \sum_{i_3=1}^{d} w_{j i_1 i_2 i_3}^{(3)} x_{i_1} x_{i_2} x_{i_3} + \cdots}_{\text{Higher order terms}}$$

Higher order terms increase the non-linear descriptive capability of the individual neurons within a neural network

Fully connected Higher Order Neural Network (HONN)



Sigma-Pi is a sparsely Connected HONN

Failures Investigated

2 groups of failures are “common” among aircraft mishaps/crashes.

- Aerodynamic Failures (A Matrix problems / lost aero surfaces, bent wings)
  - Canard Failure (0.8 to -1.75 multiplier)
- Control Failures (B Matrix problems / jammed control surfaces)

Overview of Safety Monitors

Overview of new Safety Monitors

- **Neural Net Limiter**
  - Designed to prevent high rate of change of NN commands and hard range limits
  - Failure sets Sigma Pi disengage
- **Loads Monitor**
  - Model of 40 loads locations on aircraft structure
  - If any design limit loads (DLL) are exceeded, then disengage Sigma Pi

NN Limiter

Floating Limiter

- Requirements
- Design
- Simulation validation testing
- Summary

Specific Requirements

- Acceptance criteria
  - ± 2g vertical transient limit
  - ± 0.5g lateral transient limit
  - Do not exceed specified load criteria
- Induce “worst case” D sigma pi error
  - Stay within above limits

Design Approach

- Run safety monitors in FCS at 80hz
- All inputs to safety monitors are redundant (except beta, sigma pi)
- Tripped monitors will cause a downmode from sigma pi to conventional mode with a 1 sec fader
- Causes for disengagement are instrumented on TM bus
- Safety monitor parameters are changeable from config files or recompile

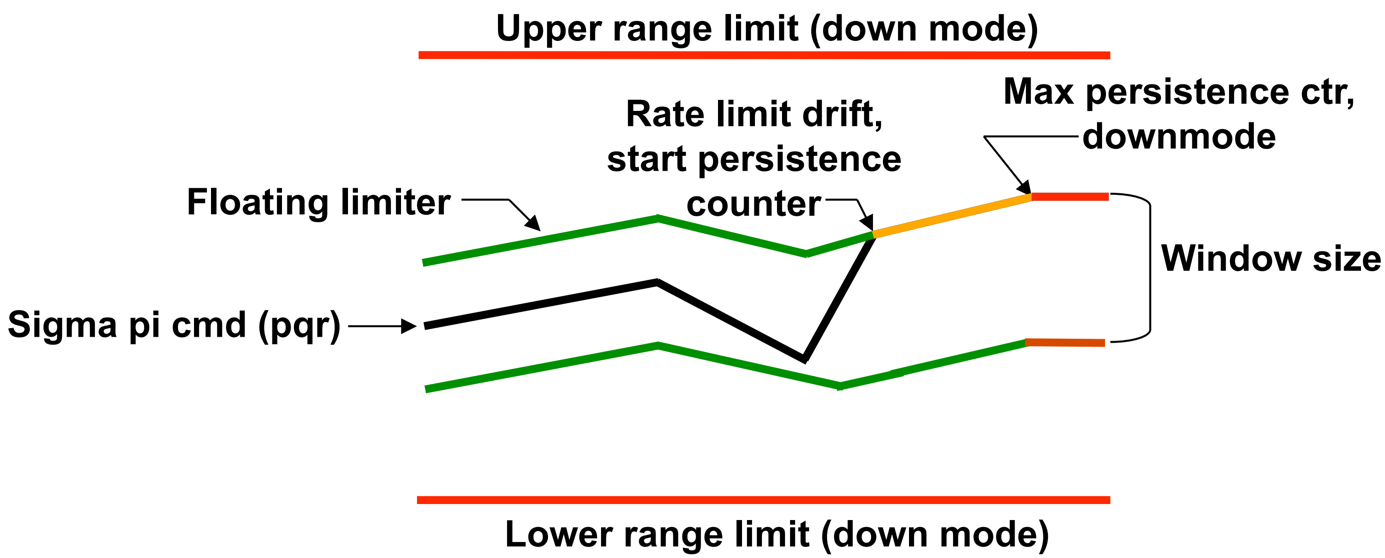
Safety Monitor Constants set from Config File Method

- **Purpose**
  - Change floating limiter or loads monitor constants without recompiling the SCE-3 code
- **Method**
  - Load config files and checksum word in ARTS using PTC and transmit to FC via 1553 bus (multiplexed)
  - FC will read data into memory and output data on FTDR bus upon command sequence from cockpit (ground operation only)
  - FC will CCDL the checksum word to all 4 channels
  - The FC will re-compute the config file checksum when the ENANCED mode is first engaged.
  - If the checksum does not match the CCDL, the ENHANCED mode will be locked out and a CONFIG fail flag will be set

Floating Limiter Design

- Apply a floating limiter window for the sigma pi commands (P,Q,R)
- Maximum rate of change is allowed within the window
- Limit the rate of change while on the floating limiter boundary
- Allow full authority up to the range limiter
- Provide flags to sigma pi to stop learning

NN Floating Limiter



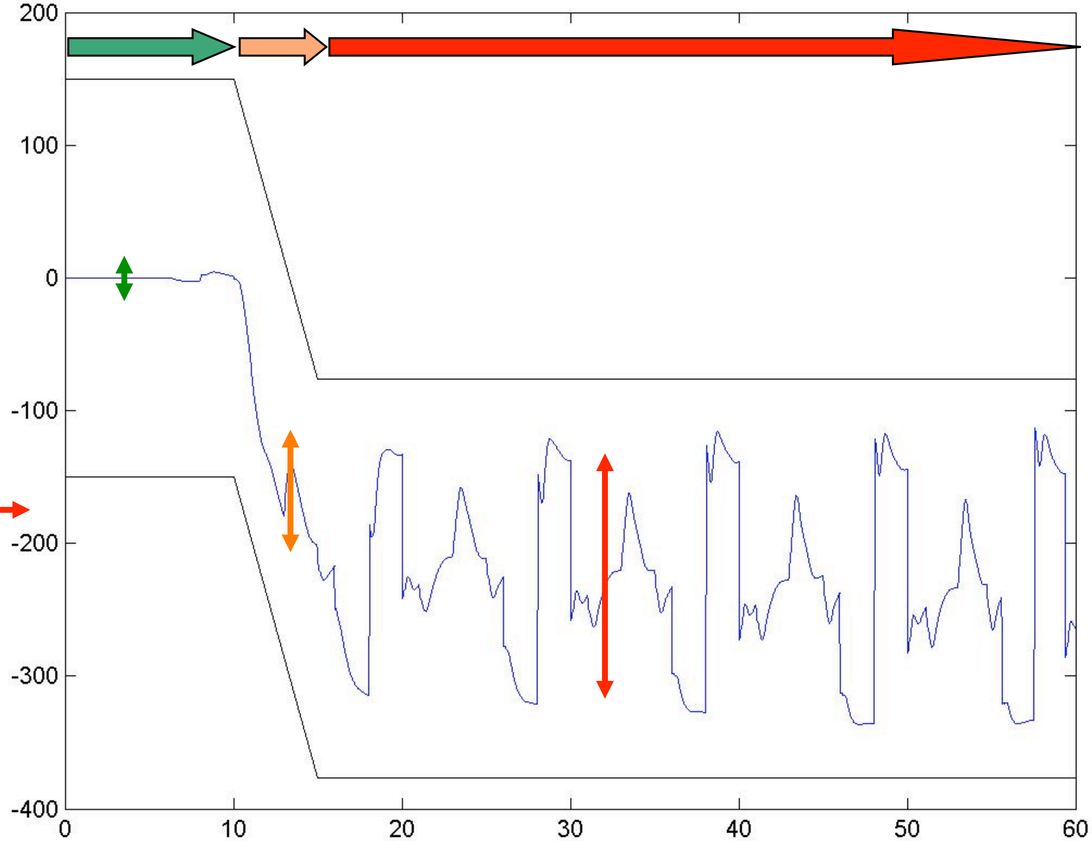
Black – sigma pi cmd
Green – floating limiter boundary
Orange – limited command (fl_drift_flag)
Red – down mode condition (fl_dmode_flag

Tunable metrics
- Window delta
- Drift rate
- Persistence limiter
- Range limits

Floating Limiter Regions

Initial; no fail
Transition
Final; after fail

Slow drift
Moderate drift
Fast drift

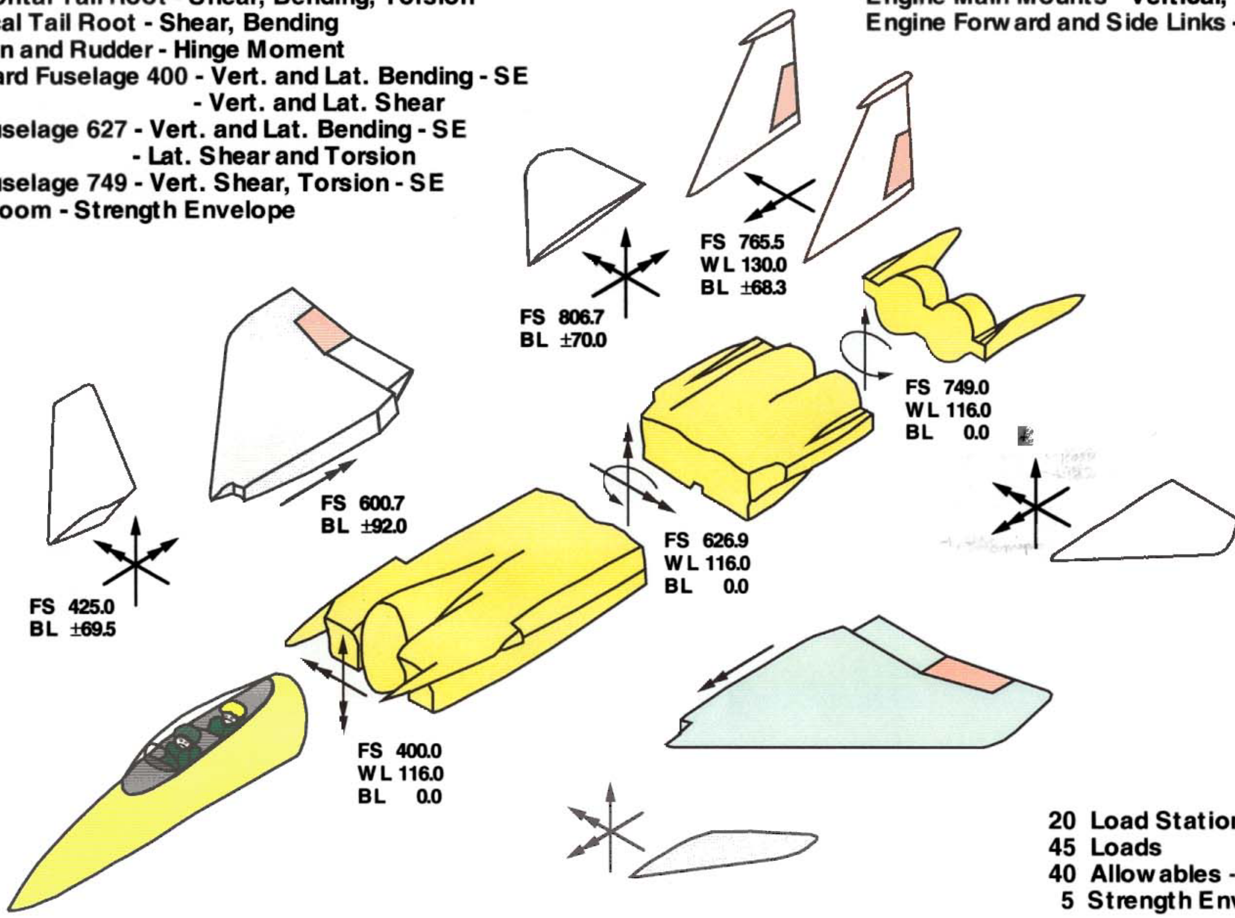


Loads Monitor

Loads Monitor Stations

Canard Root - **Shear, Bending, Torsion**
Wing Root - **Bending**
Horizontal Tail Root - **Shear, Bending, Torsion**
Vertical Tail Root - **Shear, Bending**
Aileron and Rudder - **Hinge Moment**
Forward Fuselage 400 - **Vert. and Lat. Bending - SE**
- Vert. and Lat. Shear
Aft Fuselage 627 - **Vert. and Lat. Bending - SE**
- Lat. Shear and Torsion
Aft Fuselage 749 - **Vert. Shear, Torsion - SE**
Tail Boom - **Strength Envelope**

Not Shown :
Forward Fuselage 425 - **Torsion**
Engine Main Mounts - **Vertical, Thrust - SE**
Engine Forward and Side Links - **Axial**



20 Load Stations
45 Loads
40 Allowables - %DLL
5 Strength Envelopes

Positive Forces and Moments Shown

V&V

Verification

- Simulink Block Diagrams on NN
- Define I/O of NN signals
- Test for out of range for input signals
- Test for fault detection, identification, reversion logic
- Test NN with safety monitor to limit loads and G excursions
- Test loads and floating limiters
- Document test results

Validation

- Perform Avionics System closed loop Interface test
- Perform closed loop 1553 bus I/O testing, data latency, sample rate
- Perform closed loop testing using input file from and compare results
- Evaluate transients from Adaptive to conventional mode
- Perform functional test plan & Flight Test.

Neural Network Flight Test Video

[A] matrix failure with adaptation on and off during a 1 g formation flight

Conclusions

- V&V for this project was by Limiting the size of the single string inputs from the ARTSII computer.
- We had 14 neural network trip outs due the the floating limiter.

Questions?