Hierarchical Adaptive Critics for Aircraft Control

Thaddeus Shannon, Michael Carroll and George Lendaris

Northwest Computational Intelligence Laboratory
Systems Science Ph.D. Program
Portland State University
PO Box 751, Portland, OR 97201
lendaris@sysc.pdx.edu, shannon@sysc.pdx.edu

This work is supported by the National Science Foundation under grant ECS-9904378, with additional support from NASA Langley and Accurate Automation Corporation.

Overview

- Problem Context
- DHP Method
- Fuzzy TSK Models
- Aircraft Definitions
- Decoupled – Hierarchical Approach
- Conclusion

Context

- Design of a robust flight control system for an unstable aircraft using adaptive critic methods.
- Must be capable of performing a wide range of flight maneuvers, therefore must consider a full non-linear aircraft model.
- Aircraft is particularly non-linear with respect to angle of attack (α).

DHP

Given a "primary" utility function *U(t)* that embodies a control objective at time *t*, form the "secondary" utility function

$$J(t) = \sum_{k=0}^{\infty} \gamma^k U(t+k)$$

which represents the desired control objective through time.

Adaptive critic based approximation techniques attempt to estimate *J(t)* using the identity:

$$J(t) = U(t) + \gamma J(t+1).$$

DHP Critic

In the DHP method the critic estimates the derivatives of J(t) with respect to the system states, i.e.

$$\lambda_i(t) = \frac{\partial J(t)}{\partial R_i(t)}$$

From the recursion we have

$$\frac{\partial}{\partial R_i(t)} J(t) = \frac{\partial}{\partial R_i(t)} \left(U(t) + \gamma J(t+1) \right),$$

or

$$\lambda_i(t) = \frac{\partial U(t)}{\partial R_i(t)} + \frac{\partial U(t)}{\partial u_j(t)} \frac{\partial u_j(t)}{\partial R_i(t)} + \gamma \lambda_k(t+1) \left[\frac{\partial R_k(t+1)}{\partial R_i(t)} + \frac{\partial R_k(t+1)}{\partial u_m(t)} \frac{\partial u_m(t)}{\partial R_i(t)} \right]$$

Controller Training

Controller training then utilizes the chain rule and the system model to translate critic outputs into controller training signals, i.e.

$$\frac{\partial J(t)}{\partial u_k(t)} = \frac{\partial U(t)}{\partial u_k(t)} + \gamma \sum \lambda_i(t+1) \frac{\partial R_i(t+1)}{\partial u_k(t)}$$

Fuzzy TSK Structure

Use rules of the form:

IF (u_1 is $U_{i,1}$) **AND** (u_2 is $U_{i,2}$) **AND** … **AND** (u_n is $U_{i,n}$) **THEN** $y = f_i(u)$

where $u = (u_1, u_2, \ldots, u_n)$ is the input, y is the output, $U_{i,1}, U_{i,2}, \ldots, U_{i,n}$ are the linguistic values of each input variable, and $f_i(u)$ is the consequent function of the ith rule

Firing Strength of Rules

The firing level of the *i*th rule for a particular input *u* is defined as

$$w_i(u) = U_{i,1}(u_1) \wedge U_{i,2}(u_2) \wedge \ldots \wedge U_{i,n}(u_n)$$

The output of the model is then

$$y = \frac{\sum_{i=1}^{m} w_i(u)\, f_i(u)}{\sum_{i=1}^{m} w_i(u)}$$

Normalized Firing Strength

The normalized firing strength of a rule may be defined as

$$v_i(u) = \frac{w_i(u)}{\sum_{i=1}^{m} w_i(u)}$$

in which case the output may be rewritten as

$$y = \sum_{i=1}^{m} v_i(u) f_i(u)$$

Membership Functions

with Gaussian membership functions defined on u_1, u_2, … u_n by

$$w_i\ (\bar{u}) = \exp\left(-\frac{(m_{i,1}-u_1)^2}{\sigma_1^2} - \frac{(m_{i,2}-u_2)^2}{\sigma_2^2} - \cdots - \frac{(m_{i,n}-u_n)^2}{\sigma_n^2} \right),$$

$$= \prod_{j=1}^{n} \exp\left(-\frac{(m_{i,j}-u_j)^2}{\sigma_j^2} \right)$$

where $m_{i,j}$ specify the location of the center of each rule.

Linear Consequent Functions for Critic Estimation

Given a linear time invariant plant with a stable linear control and a quadratic utility function, the secondary utility function is a quadratic form

$$J(r) = r'Pr$$

The gradient will then be a linear function

$$\nabla J = Fr$$

Definitions

- **Body Axes**
  (x,y,z) = (longitudinal, lateral, vertical)
- **Oriented to the earth via Euler Angles**
  (Ψ,Θ,Φ) = (heading, pitch, roll)
- **Rigid Body Motion Described by**
  (u, v, w) = (longitudinal, lateral, vertical) velocities -alternatively described by α, β, and V
- **Rotational Motion Described by**
  (r, q, p) = angular velocities

Mode Decoupling

There is a natural decoupling of the dynamics into longitudinal and lateral modes:

longitudinal: u,w, Θ, and q (or α, V, Θ, and q), with controls δ_e and δ_t,

lateral: v, Ψ, r,Φ, and p (or β, Ψ, r,Φ, and p), with controls δ_a and δ_r,

Steady State Flight ≡ zero acceleration

3 Cases:

- Rectilinear flight (straight line) – no rotation, represents cruising, shallow climbs or dives
- Steady turning flight (level turn) – heading changes at a constant rate while pitch and roll remain constant
- Steady symmetric pull-up – constant q

Use these cases as the basis for segmenting state space.

Decoupled – Hierarchical Approach I

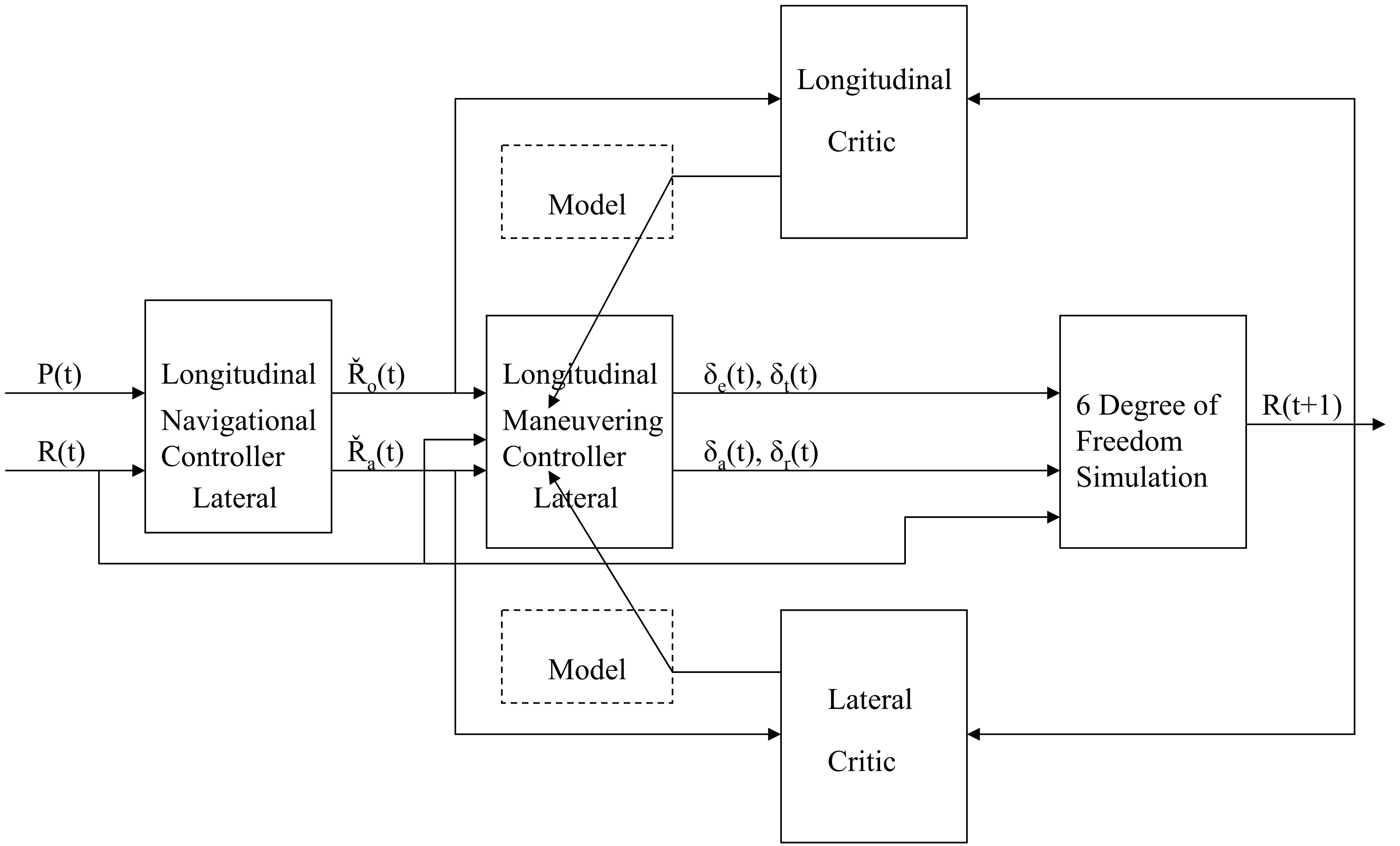
- Decompose the design problem into longitudinal and lateral controller learning systems (intended to run simultaneously).

- Pre-structure controller consequents using appropriate linear control laws for each rule's partition of state space.

Decoupled – Hierarchical Approach II

- Use individual critic estimators for each term of the utility function.
- Interleave the critic learning processes based on the relative time scales of the underlying dynamics.
- Use the resulting hierarchy of critics to train the controllers.

Controller Structure



Secondary Utility Decomposition

Define Primary Utility:

$$U(t) = \left(\alpha^* - \alpha\right)^2 + \beta^2 + \left(\gamma^* - \gamma\right)^2 + \left(\psi^* - \psi\right)^2 + \left(\varphi^* - \varphi\right)^2 + \left(V^* - V\right)^2,$$
$$= U_\alpha(t) + U_\beta(t) + U_\gamma(t) + U_\psi(t) + U_\varphi(t) + U_V(t).$$

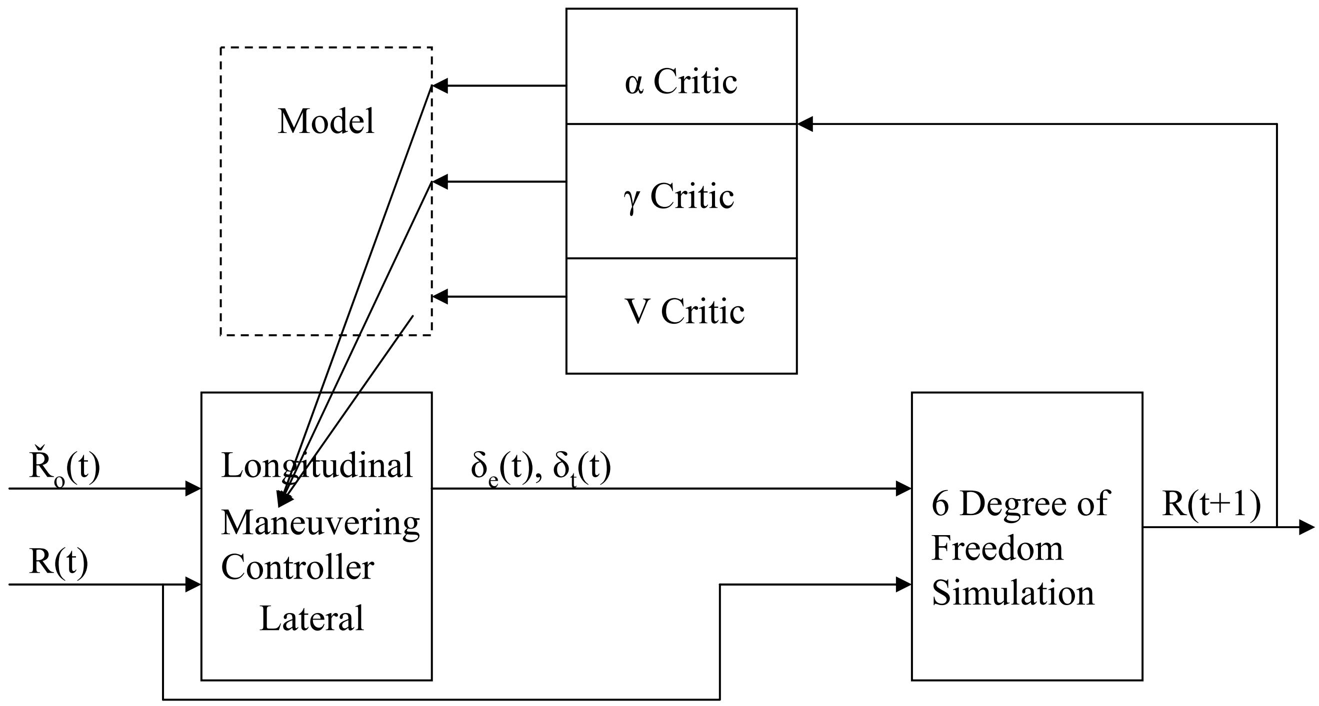
then

$$J(t) = J_\alpha(t) + J_\beta(t) + J_\gamma(t) + J_\psi(t) + J_\varphi(t) + J_V(t),$$

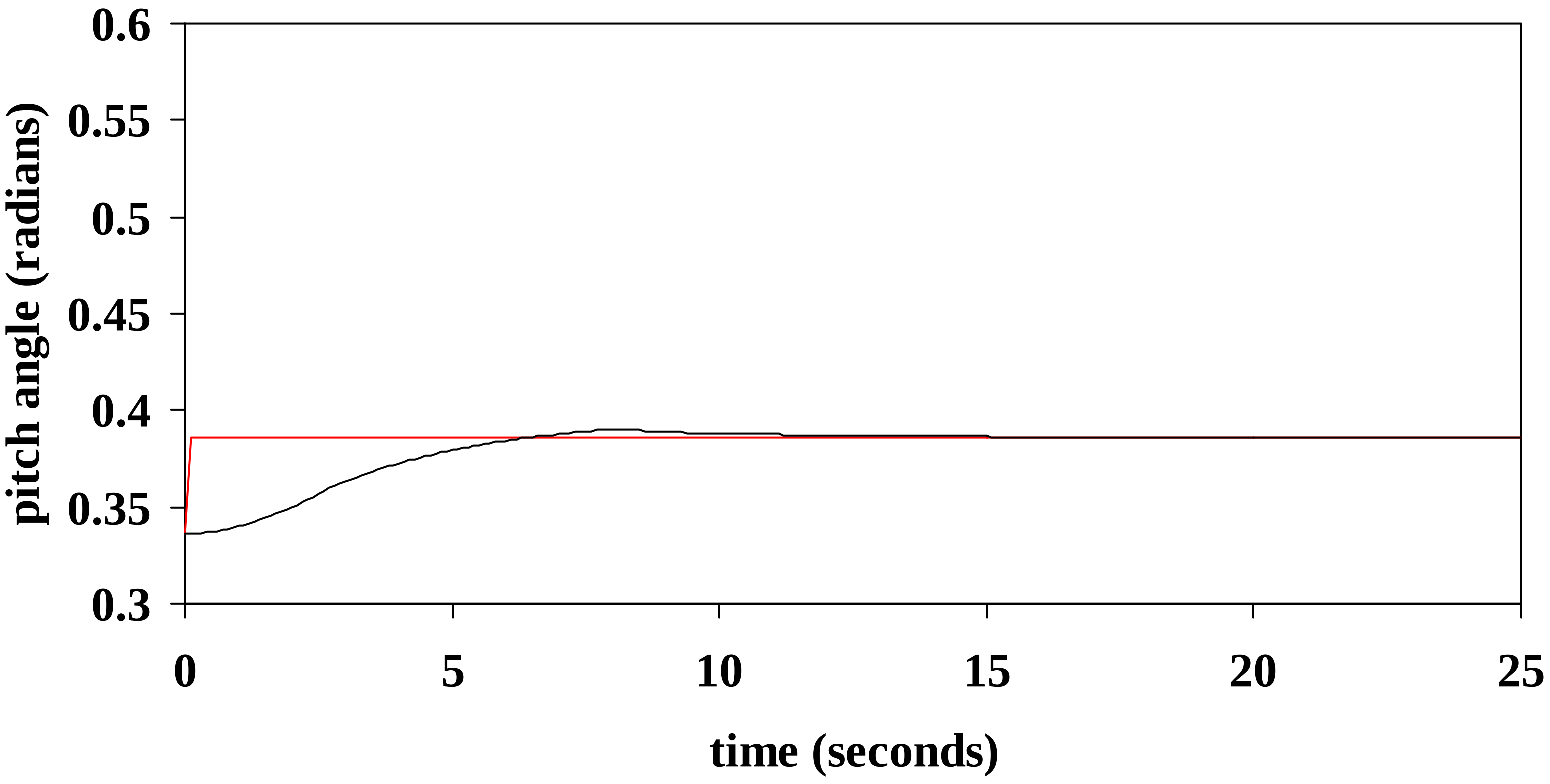
so

$$\nabla J(t) = \nabla J_\alpha(t) + \nabla J_\beta(t) + \nabla J_\gamma(t) + \nabla J_\psi(t) + \nabla J_\varphi(t) + \nabla J_V(t).$$

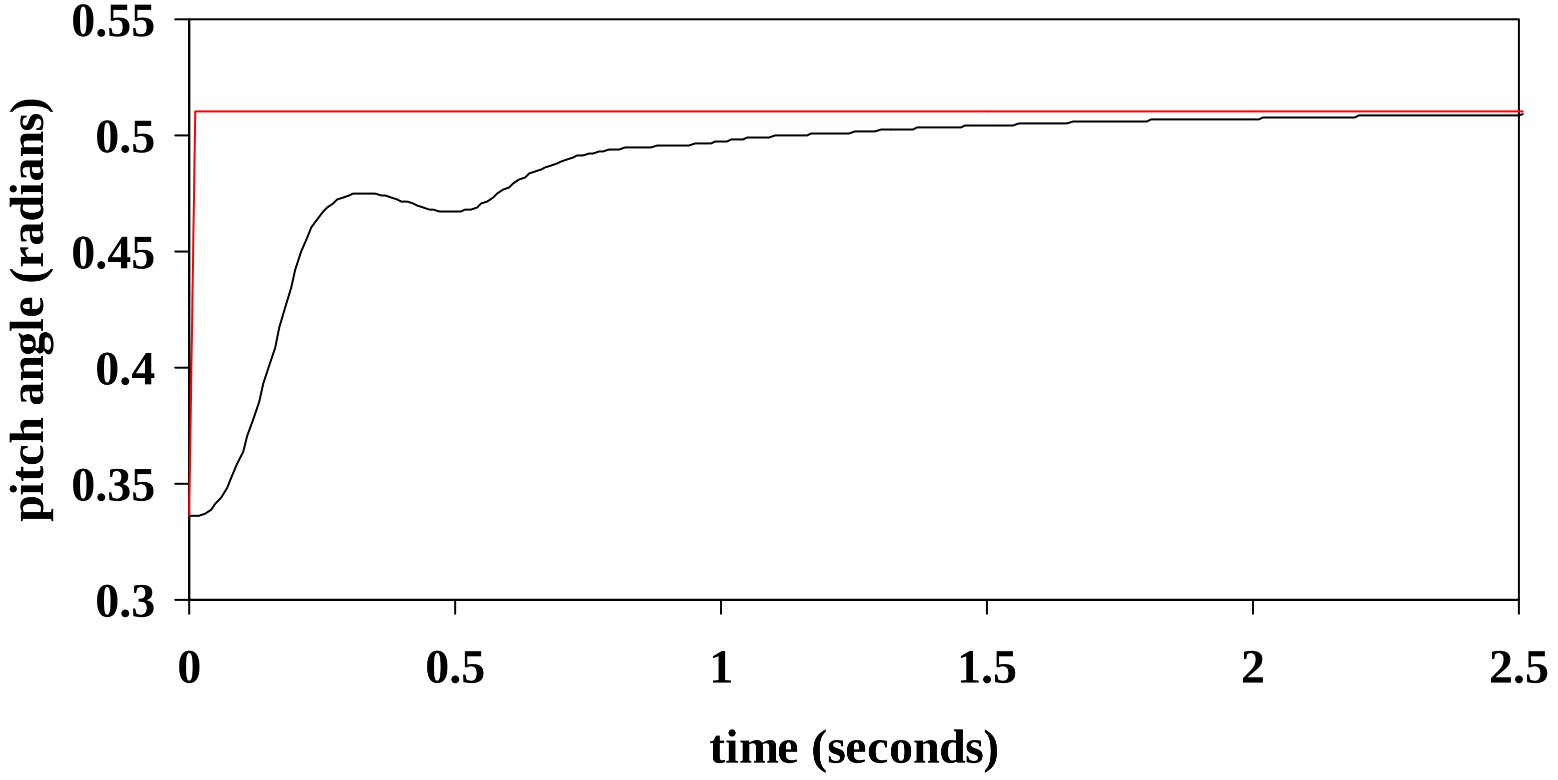
Longitudinal Learning Structure



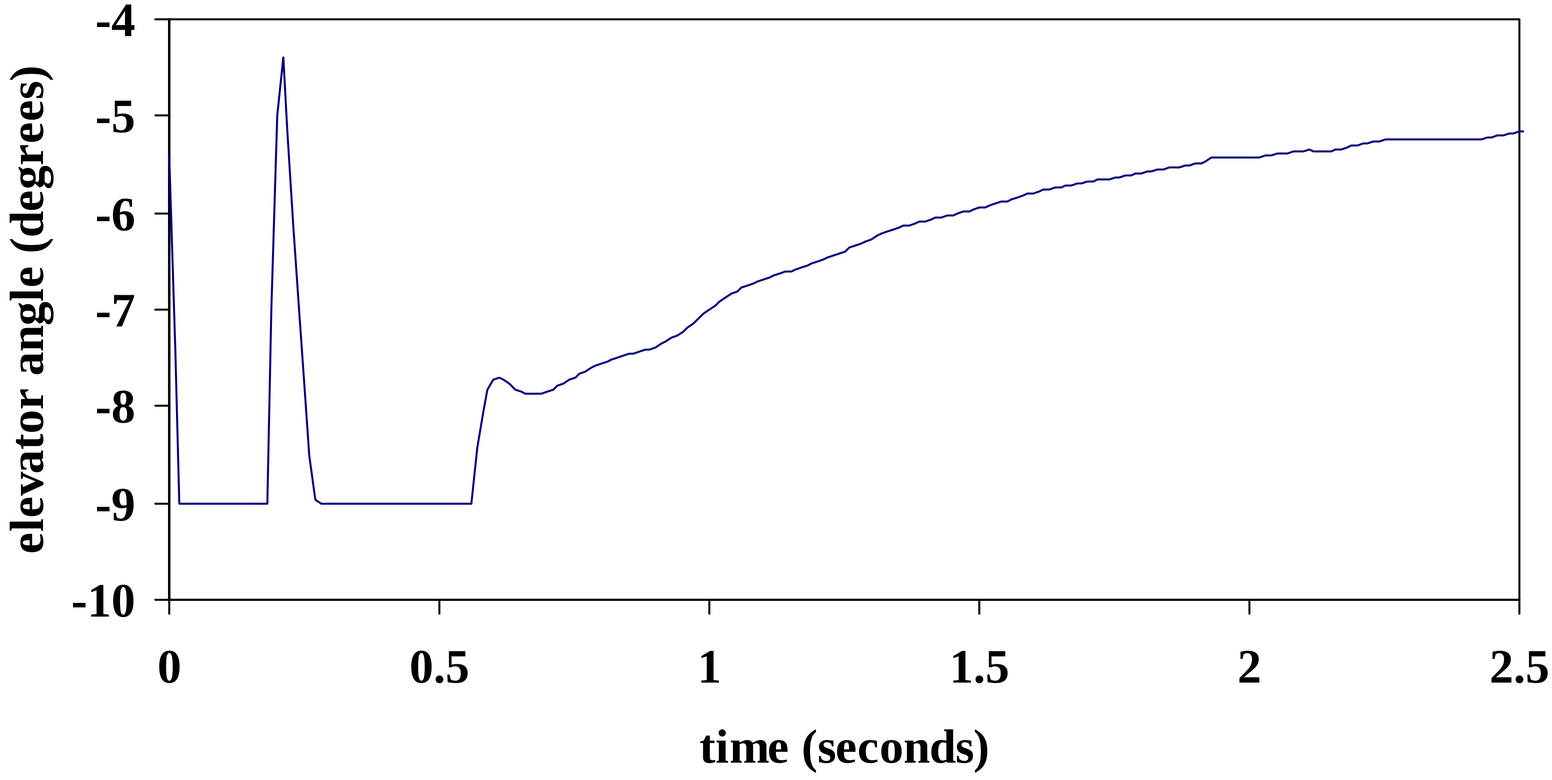
Pre Training Pitch Change Step Response



Post Training Pitch Change Step Response



Post Training Pitch Change Step Response Elevator Action



Conclusions

Take advantage of multiple objectives and time scales to simplify the design process and get:

- simpler controllers
- more parsimonious critic structures
- faster learning
- increased transparency of the design process