# Pilots Rate Augmented Generalized Predictive Control for Reconfiguration

*Don Soloway*, NASA *Ames Research Center* – don@email.arc.nasa.gov
*Pam Haley*, NASA *Langley Research Center* –p.j.haley@larc.nasa.gov

## 1. *Abstract*

The objective of this paper is to report the results from the research being conducted in reconfigurable flight controls at NASA Ames. A study was conducted with three NASA Dryden test pilots to evaluate two approaches of reconfiguring an aircraft's control system when failures occur in the control surfaces and engine. NASA Ames is investigating both a Neural Generalized Predictive Control scheme and a Neural Network based Dynamic Inverse controller. This paper highlights the Predictive Control scheme where a simple augmentation to reduce zero steady-state error led to the neural network predictor model becoming redundant for the task. Instead of using a neural network predictor model, a nominal single point linear model was used and then augmented with an error corrector. This paper shows that the Generalized Predictive Controller and the Dynamic Inverse Neural Network controller perform equally well at reconfiguration, but with less rate requirements from the actuators. Also presented are the pilot ratings for each controller for various failure scenarios and two samples of the required control actuation during reconfiguration. Finally, the paper concludes by stepping through the Generalized Predictive Control's reconfiguration process for an elevator failure.

## 2. *Introduction*

A reconfigurable flight control system redesigns the control laws in-flight in order to maintain stability and an acceptable level of handling qualities. Possible reasons for controller redesign include actuator and sensor failures, damaged or missing surfaces, and shifts in the center of gravity. A reconfigurable flight control system integrates the ability to detect and isolate failures with a system identification method to update model changes in real-time, and a means of in-flight control redesign. Some reconfigurable control designs rely on fault detection and isolation (FDI) or system identification modules to augment their reconfiguration capability [1,2,3]. The Generalized Predictive Control (GPC) system inherently incorporates all three components as part of its control process. The GPC algorithm requires a nominal linear model to initiate the prediction process, but does not require any direct knowledge of the failure. This paper demonstrates a novel use of combining the optimization algorithm, error correction, and cost function penalty terms to detect, isolate and redesign for failures without any direct measurements about the malfunction.

## 3. *Generalized Predictive Control*

The block diagram of the Generalized Predictive Control System is shown in Figure 1. It consists of four components: the aircraft that is to be controlled, an aircraft predictor model, the desired trajectory or tracking signal, ***yr***(*n*), and an optimization algorithm (Newton-Raphson) that calculates the actuator position commands needed to track ***yr***(*n*).

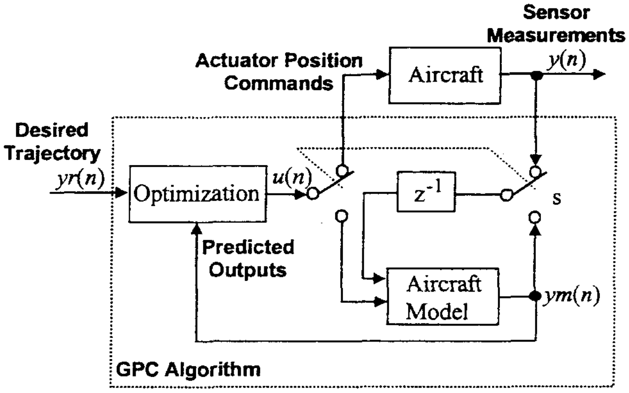

**Figure 1. Block Diagram of the GPC System**

The GPC algorithm operates in two modes, prediction and control. Prediction occurs when the switch, S, is set to the aircraft model. In this mode, the GPC algorithm uses the aircraft model to predict the aircraft's response, ***ym***(*n*), to inputs, ***u***(*n*), calculated by the optimization algorithm. When the switch is set to the aircraft, the GPC system resumes a mode of control. At this time, the control inputs, ***u***(*n*), that minimized the user specified cost function is passed to the aircraft as actuator position commands and these commands then produce the desired aircraft response, ***y***(*n*).

One of the strengths of model-based predictive control lies in the user defined cost function, which brings ease and flexibility to the control law design. Equation (1) defines the cost function designed for this application. Each of the terms in *J*(*n*) can be associated either with a cost pertaining to the aircraft's outputs or to the control inputs. Together, these terms optimize output tracking with constraints on actuator rate and position control.

$$
\begin{aligned}
\boldsymbol{J}(n) = & \sum_{i=1}^{Out} \sum_{q=\mathrm{N}_1^i}^{\mathrm{N}_2^i} \left[ yr_i(n+q) - ym_i(n+q) \right]^2 \\
& + \sum_{i=1}^{In} \sum_{q=1}^{\mathrm{N}_u^i} g_{Pos}(u_i(n+q)) + \sum_{i=1}^{In} \sum_{j=1}^{\mathrm{N}_u^i} g_{Rate}(u_i(n+q)) \qquad (1) \\
& + \sum_{i=1}^{In} \sum_{q=1}^{\mathrm{N}_u^i} g_{Sym}(u_i(n+q)) \ .
\end{aligned}
$$

The first term in ***J***(*n*) represents a cost associated with the aircraft's outputs as the sum of the squares of the error between the desired trajectory, ***yr***(*n*), and the predicted trajectory, ***ym***(*n*). *Out* is the number of outputs, $\mathrm{N}_1^i$ is the minimum-costing horizon for the $i^{th}$ output, and $\mathrm{N}_2^i$ is the maximum-costing horizon for the $i^{th}$ output. The outer summation denotes each output to be minimized while the inner summation specifies the starting and ending times for prediction, $N_1^i$ and $N_2^i$, respectively, for each output.

The remaining terms in Equation (1) represent the constraints that are placed on the inputs. The second and third terms impose constraints on each actuator's position and rate, while the last term places a penalty on certain actuators being used symmetrically.

Since a Newton-Raphson optimization algorithm requires that the function be differentiable, the second and third terms use a convex soft constraint function to account for actuator position and rate limits. The function $g_{\mathrm{Pos}}(u)$ sets the maximum and minimum deflection angle in degrees for each actuator and is defined as

$$g_{\mathrm{Pos}}(u_i(n+j)) = \frac{Ps_i}{u_i(n+j) - Plow_i} + \frac{Ps_i}{Pup_i - u_i(n+j)} \quad (2)$$
$$- \frac{4Ps_i}{Pup_i - Plow_i},$$

where
*Pup* is the positive deflection angle,
*Plow* is the negative deflection angle, and
*Ps* defines the sharpness or hardness of the function.

The function $g_{\mathrm{Rate}}(u)$ sets the upper and lower rate limit for each actuator in degrees/second and is defined using the same form as $g_{\mathrm{Pos}}(u)$ in Equation (2) [9].

The last function, $g_{\mathrm{Sym}}(u)$ places a penalty on using specific actuators symmetrically, and is defined as

$$g_{Sym}(u_i(n+j)) = \quad (3)$$
$$\lambda s_i(j)\left[u_i(n+j) + sym_i * u_{index[i]}(n+j)\right]^2,$$

where $\lambda s$ is the weighting factor, $sym = (+/-)1$ , and *index*[*i*] pairs the actuators that are to be used symmetrically. All values for the vector, $\lambda s$, are set to zero except for the actuators that are to be penalized. For example, placing a penalty on using the ailerons symmetrically will ensure that the ailerons will be used only when the elevators are not producing the desired pitch. The values for $\lambda s$, are small and can be determined by control allocation heuristics.

The outer summation on the last three terms denotes each input, *In*, to be minimized while the inner summation specifies the prediction period, $N_u$, for each input.

The Newton-Raphson optimization algorithm is a quadratically converging algorithm that requires the calculation of the Jacobian and the Hessian. Although the Newton-Raphson algorithm can be computationally expensive, the low number of iterations needed for convergence makes it a practical algorithm for real-time control [4, 5].
A complete description and derivation of the GPC algorithm using Newton-Raphson optimization for a general Single Input Single Output (SISO) system is developed in [5].

## 4. Error Correction

The GPC system was augmented with an error correction scheme that guarantees zero steady state error on the rates of the aircraft. A small tracking error in the rates will cause an aircraft to drift, which produces poor attitude-hold capabilities. The augmented GPC system is shown in Figure 2.

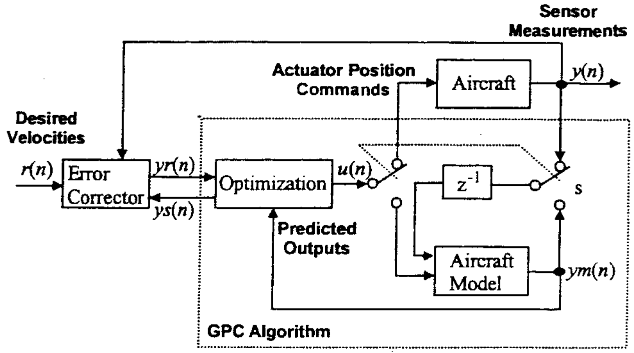

**Figure 2. GPC System With Error Corrector**

Figure 3 shows that the Error Corrector is composed from the difference between the aircraft's sensor outputs, $y(n)$, and the desired fight velocities, $r(n)$, being passed through a PI controller. This produces the reference trajectory, $yr(n)$.

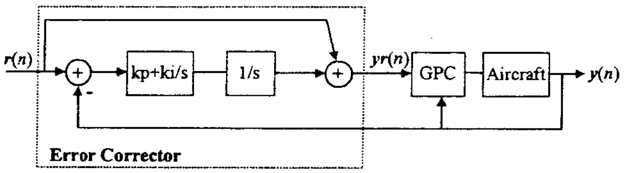

**Figure 3. Block Diagram of Error Corrector**

Another piece of the Error Corrector not depicted in Figure 3 is the anti-windup protection. Anti-windup protection avoids commands to the GPC system from growing without bounds. This could happen if a failure occurs and the actuation used for reconfiguration cannot completely compensate for the failure. This generates a residual error that accumulates in the tracking signal causing unbounded commands to be sent to the GPC Control system. This problem is avoided by monitoring the signal $ys(n)$ for commanded actuator saturation (ref. Figure 2). The vector $ys(n)$ is comprised of three binary flags (*Psat, Rsat, Ysat)* indicating the saturation state in the axis of pitch, roll, or yaw respectively. When a flag is set, anti-windup protection is engaged and integration is stopped in that axis.

For example, in this aircraft the elevators are used as the primary control of pitch. If there is a failure in the elevators, their control signal becomes saturated. GPC then reconfigures the aircraft to use the ailerons symmetrically as the secondary control for pitch. If the elevators and ailerons both saturate due to a failure or a limit in control authority then *Psat* equals 1 and anti-windup protection is employed. The pitch axis saturation flag is defined by

$$Psat = Elev_{sat} \times \left( A_{low} + |A_{roll}| \le \frac{A_l + A_r}{2} \le A_{up} - |A_{roll}| \right) \quad (4)$$

where

$Elev_{sat}$ is a binary value indicating elevator saturation,
$A_{roll}$ is the maximum degree ailerons are used for roll,
$A_{low}$ is the lower bound in degrees for the aileron,
$A_{up}$ is the upper bound in degrees on the aileron,
$A_l$ is the left aileron commanded deflection, and
$A_r$ is the right aileron commanded deflection.

The term $A_{roll}$ in Equation 4 is used when symmetric ailerons are used for secondary pitch control along with a roll command. $A_{roll}$ allows ailerons to be used for both pitch and roll commands by specifying the maximum amount differential aileron are commanded. In the current implementation $A_{roll}$ is set by hand.

This approach can be extended to propulsion and other surfaces such as flaps, spoilers and even the landing gear.

If anti-windup protection is engaged, integration along that axis stops and the states associated with the saturated axis's reference signal, $yr$(n), are set to the current state of $y$(n). Anti-windup protection is disengaged when the axis's state is less than the reference model's state.

In our last publication, a neural network was used as the predictor model for the aircraft [9]. The role of the neural network was to adapt the model to the changes in the aircraft (such as stuck actuators) thus reducing the error between the aircraft and the neural predictor model. This resulted in a reduction in the tracking error because a better model makes better predictions. In this paper the error corrector was a simple augmentation that allowed a linear model to be used in the place of the neural network. This yields the same results, but without the overhead and unknown convergence issues associated with using a neural network. There are also concerns in the FAA community about neural network based controllers being able to be validated. There are still places that a neural network may be needed for reconfiguration but for the scenarios studied here, the classical integral action of this error corrector seems to be best suited for reducing tracking error.

## 5. *Aircraft's Model and Simulation*

An expanded diagram of the aircraft's control system is shown in Figure 4. The desired aircraft's rate signals are based upon the pilot's stick commands that specify the aircrafts pitch ($\delta_{long}$), roll ($\delta_{lat}$), and yaw ($\delta_{rud}$) rates. These signals are then filtered to produce the desired handling qualities, [8], and turn coordination. Handling qualities dictate how the aircraft responds to the pilot's stick commands (i.e. agility and smoothness). Turn coordination assists the pilot by blending a yaw or roll component to the turn so that airflow is directed over the fuselage in a manner that reduces drag. From here, the signal $r(n)$ is comprised of the filtered pitch (q), roll (p), and yaw (r) rate commands. The error corrector takes $r(n)$ and the aircraft's sensor measurements for p, q, and r, $y(n)$, to produce the tracking signal, $yr(n)$. The GPC system produces the optimal actuator configuration, $u(n)$, needed to track the desired pilot commands.

The quality and effectiveness of the control input relies on the accuracy of the predictions from the model, $ym(n)$. In the predictive control scheme, the model is used to predict the future behavior of the system. In this application the model used for the entire flight envelope is a nominal linear ARMA model.

There are three types of inputs that go into the linear model: external, recurrent, and inputs that are fed back from the aircraft signals. They are $u(n)$, $ym(n-1)$, and $y(n-1)$ respectively. The external input is generated from the optimization of the cost function and results in actuator position commands. The actuators available for control are symmetric elevators, left and right ailerons and the rudder. The recurrent inputs are the estimated outputs from the last sample. The recurrent inputs are U (x-axis velocity), V (y-axis velocity), W (z-axis velocity), θ (pitch angle), and ϕ (bank angle). For initialization of the model during prediction, the aircraft's most current sensor measurements: p (pitch), q (roll), and r (yaw), are fed back.

When GPC is in the mode of prediction the switch is set to the aircraft's model. The model is initialized by setting the recurrent inputs to zero and the inputs that are fed back from the aircraft are set to the last measured values from the aircraft, $y(n\text{-}1)$. For predicting all future steps (i.e. $N_2 > 1$) both the recurrent and the fed back inputs get the values of the model, $ym$, at the previous time step. This is because there is now an estimate of the unmeasured outputs and there is only an estimate of the future measured outputs.

As mentioned earlier a single point linear model was used for all flight conditions. The flight condition was level, and trimmed at an altitude of 2000 feet, and an air speed of 197 knots. Being able to use this single model for the entire flight envelope demonstrates GPC's ability to be robust to modeling errors.

The aircraft simulation block is an extremely high-fidelity simulation model. It is the Advanced Concepts Flight Simulator (ACFS) located at the NASA Ames Crew-Vehicle Systems Research Facility (CVSRF). It is a full-mission, six degree-of-freedom, real-time motion-base flight simulator representing a generic commercial transport aircraft. The ACFS has been described as containing aerodynamics and auto-flight controls representative of a Boeing 757-class aircraft coupled with Boeing 747-400 style flight instruments [7].

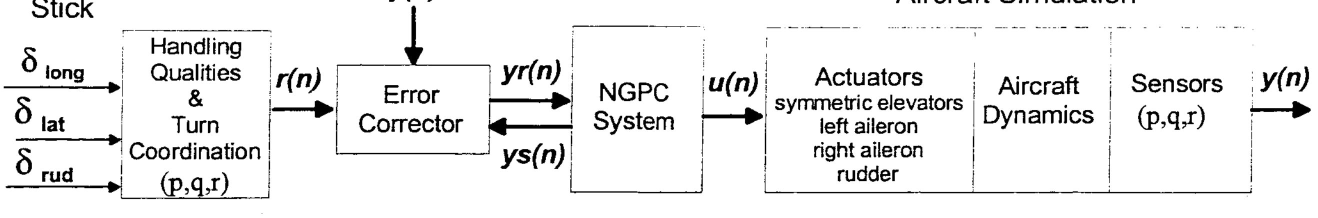

**Figure 4. Expanded Control System.**

## 6. *Piloted Ratings*

The piloted study was conducted with three Dryden test pilots. This study was to compare the handling qualities of the GPC controller and the Dynamic Inverse Neural Controller, with the conventional Stability Augmentation System (SAS). Each controller was evaluated for several different failure scenarios and was rated based on the Cooper-Harper handling qualities scale, which is summarized in Table 1 and described in detail in [11].

Four separate failure scenarios were tested under several flight maneuvers. One of these was a full tail failure, which had the rudder and both elevators frozen at 0° and the horizontal stabilizer stuck at 5° nose down. The second failure scenario was a hardover rudder (HOR) stuck at 37°. The third failure scenario was a left engine failure with the right elevator frozen at 0° that occurred during take-off. Then finally a left engine failure with the right elevator frozen at 0° and the horizontal stabilizer stuck at 5° nose down that occurred during landing.

**Table 1. Cooper-Harper Rating Scale.**

| Aircraft Characteristics | Demands on Pilot in Selected Task | Pilot Rating |
|---|---|---|
| Excellent/Highly Desirable | Pilot compensation not a factor for desired performance | 1 |
| Good/Negligible Deficiencies | Pilot compensation not a factor for desired performance | 2 |
| Fair - Some Mildly Unpleasant Deficiencies | Minimal pilot compensation required for desired performance | 3 |
| Minor, but Annoying Deficiencies | Desired performance requires moderate pilot compensation | 4 |
| Moderately Objectionable Deficiencies | Adequate performance requires considerable pilot compensation | 5 |
| Very Objectionable, but Tolerable Deficiencies | Adequate performance requires extensive pilot compensation | 6 |
| Major Deficiencies | Adequate performance not attainable with maximum tolerable pilot compensation. Controllability not in question. | 7 |
| Major Deficiencies | Considerable pilot compensation is required for control. | 8 |
| Major Deficiencies | Intense pilot compensation is required to retain control. | 9 |
| Major Deficiencies | Control will be lost during some portion of required operation | 10 |

Three types of maneuvers were tested. They were gross acquisition, landing, and takeoff. Gross acquisition is grossly maneuvering to a desired angle verses performing fine tracking to make a maneuver such as a landing. The gross acquisition (GA) was performed at an altitude of 8000 feet. with a speed of 250 knots. It consisted of obtaining each bank angle (0°, 10°, -10°, 20°, -20°, 30°, -30°, 0°) and then each pitch angle (Δ0°, Δ5°, - Δ5°, Δ10°, - Δ10°, Δ0°). The landing maneuver started 15 nautical miles from the airport at an altitude of 2000 feet with a speed of 200 knots. All the maneuvers including the takeoff were performed in winds of 20 knots from 250°, clear visibility, and light turbulence.

The pilots flew the maneuvers for each controller under both normal and failed conditions. Table 2 shows the failure scenarios along with the averaged pilot ratings for each controller. The pilots rated the handling quality of the aircraft for lateral and longitudinal control.

Overall, the pilots found that GPC and the Dynamic Inverse Neural controller were similar when looking at how the aircraft handles during and after reconfiguration. One difference between the controllers can be seen when looking at the requirements placed on the various control surfaces in order to achieve reconfiguration. The next section takes a closer look at this difference.

**Table 2. Pilot Survey of Failure Scenarios.**

| Controller | Failure | Maneuver | Lat. Rating | Long. Rating |
|---|---|---|---|---|
| SAS | none | GA | 2 | 2 |
| DINV | none | GA | 2 | 3 |
| GPC | none | GA | 2 | 1 |
| SAS | none | Land | 3 | 3 |
| DINV | none | Land | 3 | 3 |
| GPC | none | Land | 3 | 3 |
| SAS | none | Take Off | 2 | 2 |
| DINV | none | Take Off | 2 | 2 |
| GPC | none | Take Off | 2 | 2 |
| SAS | Tail Fail | GA | 8 | 8 |
| DINV | Tail Fail | GA | 4 | 5 |
| GPC | Tail Fail | GA | 3 | 4 |
| SAS | Tail Fail | Land | 9 | 10 |
| DINV | Tail Fail | Land | 5 | 5 |
| GPC | Tail Fail | Land | 5 | 5 |
| SAS | HOR | GA | 5 | 3 |
| DINV | HOR | GA | 5 | 2 |
| GPC | HOR | GA | 6 | 2 |
| SAS | HOR | Land | 7 | 2 |
| DINV | HOR | Land | 5 | 2 |
| GPC | HOR | Land | 5 | 2 |
| SAS | Eng Out | Take Off | 10 | 8 |
| DINV | Eng Out | Take Off | 7 | 5 |
| GPC | Eng Out | Take Off | 7 | 5 |
| SAS | Eng Out | Land | 10 | 8 |
| DINV | Eng Out | Land | 8 | 5 |
| GPC | Eng Out | Land | 7 | 5 |

## 7. *Actuation Requirements*

Handling qualities are a measure of how well the control system allows a pilot to perform a particular mission or task and to what level of ease and precision is it managed. However, this does not look at the demand that is required from the control surfaces and their actuators to obtain that level of control. Actuation requirements are a concern when high rates are needed and the hardware is not designed for those conditions. This could shorten the actuators life or cause additional failures. Ideally the control system would reconfigure with the minimal requirements on the surface positions and actuator rates. For most of the failure scenarios the data shows that GPC's requirement for the control surfaces and actuator rates were less than with the DINV controller. One example of this is shown in Figures 6, 7, and 8 where both controller's positions and rates are compared for the aileron, elevator, and rudder respectively. The data shown is for the last failure scenario in Table 2 where the left engine is dead and the right elevator and horizontal stabilizer are frozen while attempting a landing. Keep in mind that we want to minimize the sum total requirements for all actuators simultaneously. You can see that overall, GPC does not actuate the surfaces as much as DINV.

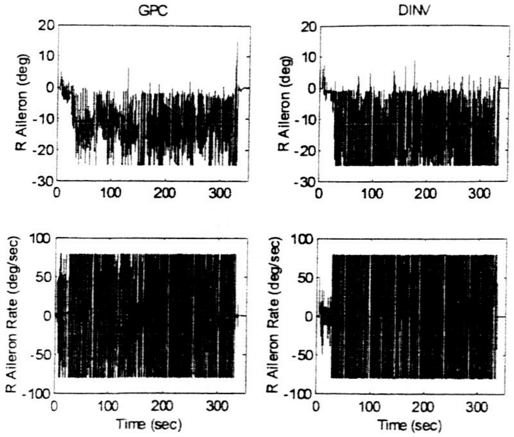

Figure 5. Aileron surface Position and Rates

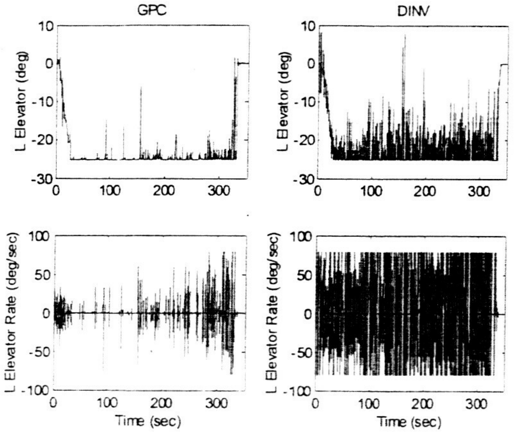

Figure 6. Left Elevator surface Position and Rates

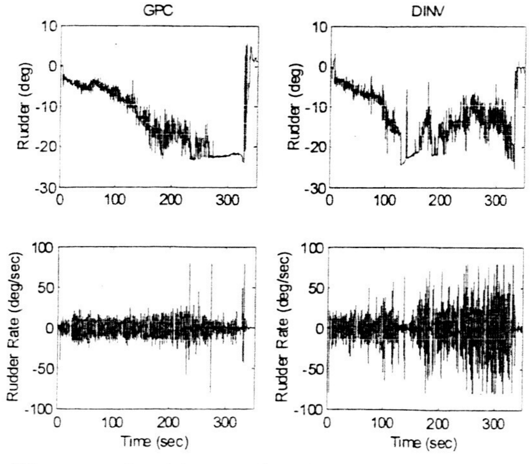

Figure 7. Rudder surface Position and Rates

To compare the actuator usage between the controllers a RMS measure for each actuator rate was calculated. For example, from Figure 6 the RMS for the aileron is 0.01407 for GPC as compared to 0.02024 for DINV. A summary of several test cases is shown in Table 3. Only the test cases that were comparable are included. For example, all GA maneuvers are excluded because they are difficult to compare. Here you can see that for all cases except the first one, GPC had less actuation than DINV. Even though neither controller was tuned to minimize actuator usage, we suspect that GPC has an advantage since the cost function includes both position and rate constraints in the optimization.

**Table 3. RMS of the Rates for each Surface**

| Control | Failure | Test | Aileron | Elev | Rud |
|---|---|---|---|---|---|
| GPC | none | land | 0.00426 | 0.00484 | 0.00153 |
| DINV | none | land | 0.00223 | 0.00460 | 0.00237 |
| GPC | none | TO | 0.01296 | 0.00664 | 0.00537 |
| DINV | none | TO | 0.01852 | 0.02615 | 0.00619 |
| GPC | tail | land | 0.01714 | 0.00011 | 0.00004 |
| DINV | tail | land | 0.01940 | 0.00133 | 0.00035 |
| GPC | hor | land | 0.01511 | 0.00325 | 0.00320 |
| DINV | hor | land | 0.02021 | 0.00934 | 0.00532 |
| GPC | eng out | land | 0.01407 | 0.00267 | 0.00242 |
| DINV | eng out | land | 0.02024 | 0.00940 | 0.00446 |

## 8. *A Closer Look at GPC Reconfiguration*

This section describes in more detail a failure scenario where GPC compensates for a full tail failure. The purpose is to step through the failure and describe how the GPC algorithm and its cost function work together to accommodated the failure. The aircraft is descending from 4000 to 2000 feet when the full tail failure occurs. Figure 8 compares the SAS and GPC controller for this failure scenario. Around 140 seconds into the maneuver the failure occurs and GPC engages symmetric ailerons to compensate for the loss of pitch control. The standard SAS controller cannot reconfigure, so control is lost. The pilots rated the longitudinal control of the SAS controller with a 10 and the reconfigurable controllers with a 5.

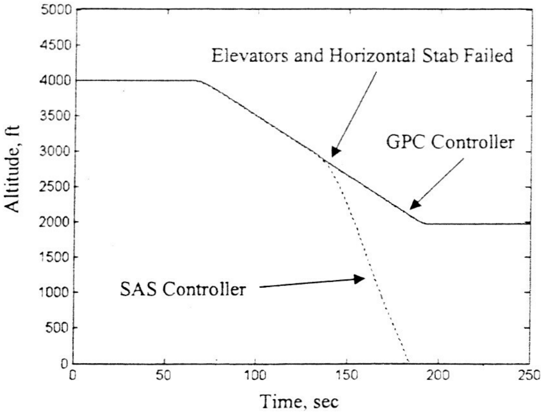

Figure 8. Comparison of SAS and GPC

The commanded deflection of the elevator verses the ailerons during reconfiguration are shown in Figure 9. The dotted line is the commanded elevator deflection and the solid represents the commanded symmetric aileron deflection. The start of the maneuver occurs at 80 seconds when the aircraft was commanded to pitch down. GPC commanded the elevators down to track the pitch command. At 140 seconds the elevator failed at 0° deflection. This started to cause the aircraft to level out causing GPC to command more elevators until their limit was reached. Since the elevators were saturated with no change in pitch rate, GPC switched from commanding elevators to commanding symmetric ailerons. The desired altitude was reached within 200 seconds.

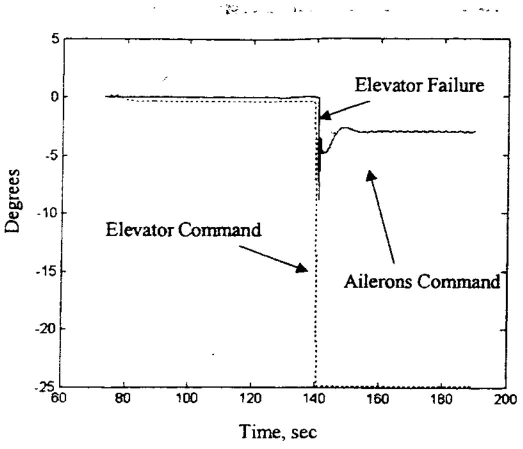

**Figure 9. Elevator and Aileron Deflections**

Each of the terms in the cost function affects how GPC responds to pilot commands and reconfiguration needs. For example, when the pilot commands a change in altitude, GPC calculates an elevator deflection that will track the pitch rate command. The commanded elevator deflection minimizes the tracking error (the first term in Equation (1)). The costs associated with the position, rate, and $g_{\text{sym}}(u)$ constraints (the third, fourth, and fifth terms in Equation (1)) are zero since saturation has not yet occurred and ailerons have not been engaged. When the elevator fails, the tracking error starts to increase resulting in an increase in $J(n)$. GPC continues to command the elevator until the cost associated with the use of the elevators becomes too high (when the elevators approach the saturation limits imposed by $g_{\text{Pos}}(u)$ in Equation (1)). When this occurs, an increase in the cost of $g_{\text{sym}}(u)$ is allowed, and the use of the ailerons symmetrically is accepted.

## *9. Conclusions*

This paper offers an alternative reconfigurable control approach that performs tracking tasks as well as the neural based Dynamic Inverse controller. The Generalized Predictive Controller minimized the rate requirements on the actuators better than the Dynamic Inverse controller. This is thought to be due to the inherent ability of predictive control to include constraints on the rates during the optimization.

Another advantage of this controller is that equal performance was obtained without the use of a neural network. Neural networks are great at adapting to unmodeled dynamics or a model that is changing, but currently, there is no way to validate a neural network's response unless it's inputs and outputs are bounded in some way. Using a neural network adds an unnecessary uncertainty that makes near-term FAA acceptance highly unlikely.

In extreme cases where the aircraft has dramatically changed, such as a partial or total loss of a wing, a neural network or another form of model update will be necessary. This is when the aircraft has changed so much that any form of error correction is not enough. In this case, the predictive controller's neural network modeling capability can be enabled to adapt to the new model.

## *References*

[1] J. Monaco, D. Ward, R. Bird, "Implementation and Flight Test Assessment of an Adaptive Reconfigurable Flight Control System" *Proc. AIAA Guidance, Navigation, and Control Conference*, New Orleans, LA, AIAA-97-3738, Aug. 1997.

[2] A. Calise, S. Lee, M. Sharma, "Direct Adaptive Reconfigurable Control of a Tailless Fighter Aircraft", *Proc. AIAA Guidance, Navigation, and Control Conference*, Boston, MA, AIAA-98-4108, Aug. 1998.

[3] B. Bacon, A. Ostroff, "Reconfigurable Flight Control Using Nonlinear Dynamic Inversion With a Special Accelerometer Implementation", *Proc. AIAA Guidance, Navigation, and Control Conference*, Denver, CO, AIAA-2000-4565, Aug. 2000.

[4] P. Haley, D. Soloway, B. Gold, "Real-time Adaptive Control Using Neural Generalized Predictive Control", *Proc. American Control Conference*, San Deigo, CA, June 1997.

[5] D. Soloway and P. Haley, "Neural Generalized Predictive Control: A Newton-Raphson Implementation," *Proceedings of the IEEE CCA/ISIC/CACSD*, IEEE Paper No. ISIAC-TA5.2, Sept. 15-18, 1996.

[6] D. Soloway, "Neural Generalized Predictive Control for Real-Time Control," Masters Thesis, Old Dominion University August 1996

[7] M. Blake, "The NASA Advanced Concepts Flight Simulator: A Unique Transport Aircraft Research Environment", *AIAA* Paper 96-3518, 1996.

[8] J. Kaneshige and K. Gundy-Burlet, "Integrated Neural Flight and Propulsion Control System," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Montreal, Canada, Aug. 6-9, 2001, AIAA-2001-4386.

[9] D. Soloway, P. Haley, "Aircraft Reconfiguration Using Neural Generalized Predictive Control", *Proc. American Control Conference*, Arlington, VA, June 2001.

[10] A. Kelkar, D. Soloway, "Guaranteed Satiability of GPC with Control Allocation", *to be published Proc. American Control Conference*, Denver, CO, June 2003.

[11] G. Cooper, R. Harper Jr., "*The Use of Pilot Rating in the Evaluation of Aircraft Handling Qualities*", NASA TN D-5253, 1969.