NASA
Technical Memorandum 106859

Army Research Laboratory
Technical Report ARL–TR–906

A Method for Exploiting Redundancy to Accommodate Actuator Limits in Multivariable Systems

Jonathan Litt
Vehicle Propulsion Directorate
U.S. Army Research Laboratory
Lewis Research Center
Cleveland, Ohio

and

Greg Roulette
Shaker Heights High School
Shaker Heights, Ohio

Prepared for the
19th Science Conference
sponsored by the U.S. Army
Orlando, Florida, June 20–24, 1994

National Aeronautics and
Space Administration

U.S. ARMY
RESEARCH LABORATORY

A Method for Exploiting Redundancy to Accommodate Actuator Limits in Multivariable Systems

Jonathan Litt
Vehicle Propulsion Directorate
Army Research Laboratory
21000 Brookpark Road
Cleveland, OH 44135-3191

Greg Roulette*
Shaker Heights High School
15911 Aldersyde Drive
Shaker Heights, OH 44120-2599

ABSTRACT

This paper introduces a new method for accommodating actuator saturation in a multivariable system with actuator redundancy. Actuator saturation can cause significant deterioration in control system performance because unmet demand may result in sluggish transients and oscillations in response to setpoint changes. To help compensate for this problem, a technique has been developed which takes advantage of redundancy in multivariable systems to redistribute the unmet control demand over the remaining useful effectors. This method is not a redesign procedure, rather it modifies commands to the unlimited effectors to compensate for those which are limited, thereby exploiting the built-in redundancy. The original commands are modified by the increments due to unmet demand, but when a saturated effector comes off its limit, the incremental commands disappear and the original unmodified controller remains intact. This scheme provides a smooth transition between saturated and unsaturated modes as it divides up the unmet requirement over any available actuators. This way, if there is sufficiently redundant control authority, performance can be maintained.

INTRODUCTION

A standard linearized multivariable system is of the form

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{1}$$

where x is the vector of state variables, u is the set of control inputs, y is the vector of system outputs, and A, B, C, and D are matrices of the appropriate size. Generally, some type of linear control scheme is designed for it which divides up control responsibility among the actuators. When an actuator saturates, the linear controller may act in a nonlinear manner and abnormal performance can result. This performance may be unacceptable from a safety, cost, or quality standpoint.

Problems Associated with Actuator Saturation

An actuator is a physical system so its output, which under normal conditions is a function of its input, is restricted to lie within some boundaries. At the edge of its range of motion, additional input to drive it past the endpoint will have no effect. Thus, even though the control system is demanding more actuation, the effector cannot provide it. Additionally, because of power and wear considerations, the actuator must not be forced against its limit for extended periods. In order to assure this, the command to it must be limited so that it never tries to drive the actuator outside of its unrestricted range. As long as an actuator command stays within the normal bounds, there is no difference between the desired and achievable actuator positions. However, as an actuator command moves beyond the normal range it gets clipped, indicating that a portion of the control demand can not be met. When the control signal is limited like this, it appears to the plant that the demand is smaller than it actually is, i.e. that a component of u is smaller, resulting in a sluggish response or an error as the system is driven by the limited rather than the actual demand.

This brings up the issue of *directionality* where the vector u is more important than its individual elements. When a component of a vector is changed, the vector's direction is modified. In the geometric representation shown in figure 1, the

*Summer High School Apprenticeship Research Program participant at Lewis Research Center.

limiting of an actuator corresponds to the vector addition of the desired control vector, u_{un}, and the Δu vector which lies parallel to the axis of the saturated variable. The resultant vector, u_L, is the actual control signal reaching the plant. This may have serious consequences, especially in a decoupled control system where multiple actuators are coordinated to produce a response in a single output variable only. Modifications to the coordination may cause the decoupling scheme to be upset and may produce undesirable perturbations in the output variables.

The sluggish response and steady state error due to limited effectors is exacerbated by the use of an integral term in the control system. An integral term is used to eliminate steady state error in the closed loop system but, when the plant responds more slowly than expected due to a saturated actuator, the integral component of the demand tends to become excessively large, resulting in a phenomenon known as integral windup. Windup pushes the closed loop system toward instability because the integral term must unwind before the actuator can move off its limit, creating a delay which produces overshoot and oscillations. Much research exists on this aspect of the problem but these solutions attempt to prevent windup and do little to address the cause of the problem, unmet demand. The techniques that do try to compensate for saturated actuators can be difficult to calculate and are generally optimized for a single limited effector.

Therefore, in order to compensate for the unmet demand due to multiple saturated actuators, a new, simple scheme will be proposed. To be able to use the proposed scheme, redundant actuators must be present. This simply means that at least one nonzero component of the *Bu* vector from (1) for $u=e_i$ (a column vector of zeroes with a 1 at the *i*th location) should be nonzero when $u=e_j$, $j\neq i$. More formally, $|e_i^T B^T||Be_j|\neq 0$ when $i\neq j$. In practical terms, it implies that more than one actuator has an affect on a given variable allowing the value of that variable to be manipulated in more than one way. This is extremely common in multiple input-multiple output systems and should not be considered a limitation.

Some background issues remain to be discussed before the new algorithm is introduced along with an example and conclusions. First, some existing methods for control redistribution will be presented. Then general guidelines for integrator windup protection schemes, followed by several existing techniques both without and with redistribution capability will be described briefly.

Control Redistribution Schemes

It is important to realize that designers are not necessarily forced to live with the sluggish response due to actuator saturation. When redundancy is present other actuators can be called upon to help provide the necessary control effort. The *control mixer* is a redistribution strategy most often used for control reconfiguration after actuator damage [1,2,3]. In these cases, the control mixer matrix, K_M, which distributes the commands among the actuators, is recomputed to compensate for the impaired *B* matrix using the generalized inverse of *B* [4] in a attempt to do model matching using state feedback, i.e. $BK_M=B_{impaired}K_{Reconfigured}$. With redundancy among the actuators, the generic command tracking might be within design specifications even with impaired actuators. Actuator saturation can be considered a special case of actuator impairment with the effector movement restricted in one direction only. Redistribution of the undelivered control demand over the remaining actuators in response to a limit situation has been attempted with some success in a multivariable system [5]. The unmet request is split among the available redundant actuators using a generalized inverse of the *B* matrix to add an increment to each remaining effector. However, when the incremental demand causes a second actuator to saturate, the offending increment is removed and redistributed, resulting in spikes in the actuator commands.

Integral Windup Protection Schemes

Some basic guidelines for windup protection algorithms exist [6]. For instance, windup protection schemes should only affect the controller's output during saturation, not during the unlimited operation, i.e. they should be memoryless. When in use, the schemes should not make the controller unstable nor should they make the closed loop system unstable for any limit combination. This does not mean that a general theoretical stability proof must be developed in order to use the technique, only that the particular application with windup protection in place must be analyzed for stability in each possible limit situation. Also, these schemes should provide a smooth transition or *bumpless transfer* between limited and unlimited modes. This usually involves having the controller output track the actuator limit value during saturation. Finally, where possible the schemes should maintain system performance, but if not possible, they should bring the system to some minimally degraded condition. Thus, to reiterate, there are four main criteria integrator windup protection schemes should meet: (1) they should be memoryless, (2) they should maintain controller and closed loop system stability for any limit combination, (3) they should provide bumpless transfer, and (4) they should accommodate graceful degradation.

A quick examination of these guidelines reveals that the schemes should be proportional in nature, i.e. they should contain no dynamics of their own; and that the controller output should track the actuator limit during saturation. The stability requirement is straightforward and the graceful degradation specification is arbitrarily defined. Assuming that the scheme will met the criteria, the designer has freedom to choose the gains in terms of both limit tracking and post-saturation performance. Since all schemes will provide actuator limit tracking, it is important to choose one which is appropriate for the highly application-dependent degradation aspect of the problem.

There is one typical structure used in most windup protection schemes. It is shown in figure 2. In this arrangement, the actuator overdemand is fed back to the integrator to modify the state of the controller. Several widely different schemes will be described to demonstrate the versatility of this structure.

In single-input single-output systems, or ones where there is no redundancy, the simplest way for the controller output to track the actuator limit is to truncate the integrated error at the saturation value. The Conditioned Control [7] achieves this result without resorting to nonlinear elements in the controller. The true actuator output is subtracted from the ideal (unrestricted) actuator output. Any difference is attributable to saturation. This difference is multiplied by a gain and fed back to the input of the integrator which effectively converts the integrator to a first-order lag whose steady state value will keep the actuator at its limit. Thus, the integrator's output can not wind up and the closed loop analysis can be performed using linear methods.

Two anti-windup schemes for multivariable systems are the High Gain or Conventional Anti-Windup (CAW) approach and the Modified Anti-Windup (MAW) approach [8]. The CAW scheme has the same structure as the Conditioned Control applied to multivariable systems, i.e. the actuator overdemand is multiplied by a fixed gain and fed back to the integrator, modifying the state of the controller. The gain is selected so that the command to the actuator will be driven to its limit value quickly once that actuator saturates. The MAW scheme, which has the same structure as the CAW scheme, handles the problem of directionality as well as integral windup protection. The MAW scheme scales back the commands to the unlimited actuators so that they are proportional to the limited one and the direction of the input vector remains correct albeit smaller in magnitude than in the ideal case. This approach considers only the input vector and not its resulting effect on the system. Performance might be better maintained by utilizing inherent actuator redundancy, if it exists, rather than simple command scaling.

Yet another technique which uses the same structure minimizes the weighted squared error in both output command tracking and actuator command tracking [9]. The main objective of this method is to provide windup protection, but where appropriate the outputs of nonlimited actuators are altered through manipulation of the controller state to give a better response in terms of minimization of the objective function. This fixed-gain technique is optimized for a single limited actuator only but the optimization algorithm guarantees controller and closed loop stability for any limit combination, if possible, through the use of a brute force parameter search strategy.

A new technique [10] which also shares the same structure mentioned above tries to produce a set of static gains which minimize the L_2-norm of the limited system subject to a linear matrix inequality [11]. This constraint guarantees absolute stability for any limit combination while attempting to force the system to meet certain performance criteria. The problem formulation used to obtain the gains does not explicitly. specify anti-windup behavior even though it usually results naturally in the solution. However, in an example where actuator redundancy existed, rather than the windup being attenuated, it was *amplified* and then fed back to influence the redundant actuator resulting in the maintenance of performance. This led to the conclusion that *high gain limit protection might not be necessary or even desirable* in certain situations.

The above schemes were presented from oldest to most recent. The increase in complexity of the gain determination procedures with time is obvious. It seems apparent that as the algorithms become more complicated, specification of the objective function and constraints becomes more involved, sometimes producing totally unexpected results. It is also clear that, when performance is an optimization criterion and redundancy is available, these windup protection techniques begin to merge with the redistribution ideas presented in the previous section.

THE WINDUP FEEDBACK SCHEME

Building on the idea that windup is not bad if it can be used to improve performance, the Windup Feedback scheme is proposed. While this new technique is similar to one previously described [5], in the new method an incremental command which causes actuator saturation is not removed. Rather, this unmet control demand is again fed back and distributed over the remaining unsaturated actuators as long as redundancy exists. Thus the proposed system acts as a control effort *shunt* or *diverter*, redirecting the unmet demand away from the saturated actuator to the redundant effectors, to which it appears as an incremental command beyond that supplied by the multivariable controller.

A block diagram of the Windup Feedback scheme is shown in figure 3. Note how it differs from the standard structure for windup protection where the overdemand is fed back to affect the controller state. Here the extra demand is redistributed directly to the redundant actuators. In this setup, generalized actuator commands—commands for each orthogonal control input no matter how many physical actuators realize it—are used. The feedback variable, y_H, is subtracted from the generic setpoint vector, *r*, to produce an error vector, *e*. This error vector enters the control system, $G_C(s)$, and the generalized control vector, e_r, results. It passes through the designer-specified transformation or *mixer*, K_M, which splits up and distributes the generalized command vector into an actual command vector, u_{UN}. The new vector gets clipped which assures that its output, u_L, is within range and will not force any actuator hard against its limit. The bounded command signals drive the actuators, $G_a(s)$. The actuator output vector, u_a, then enters the plant, *G(s)*, and the resulting output vector is *y*. This system output vector enters a block in the feedback path, *H(s)*, creating the vector y_H, which is fed back.

It is the unmet demand Δu, the difference between u_{UN} and u_L, which is used to compute the Windup Feedback. The Windup Feedback is allocated among the remaining actuators in such a way that the derivatives of the state variables agree as closely as possible with those of the same system with unlimited effectors. This is achieved through Windup Feedback gains whose values are chosen to minimize, in a least squares sense, the difference between the control contribution to the limited and unlimited systems' state variables. In other words, the difference between the control term for the ideal, unlimited system and the limited but compensated system is minimized. Thus, with variables defined as in figure 3, the objective function is specified as

$$\begin{aligned} J &= \frac{1}{2}\,[B u_{UN} - B(u_{UN} - \Delta u + I^* I^{*T} u^*)]^T Q\,[B u_{UN} - B(u_{UN} - \Delta u + I^* I^{*T} u^*)] \\ &= \frac{1}{2}\,[B(\Delta u - I^* I^{*T} u^*)]^T Q\,[B(\Delta u - I^* I^{*T} u^*)] \end{aligned} \qquad (2)$$

where I^* restricts the optimized feedback matrix to utilize only unsaturated actuator commands. I^* is created by taking the identity matrix of dimension equal to the number of actuators and deleting each column which corresponds to a saturated actuator command. This way, whenever a limit is encountered, I^* is computed to be the dimension of the total number of actuators by the total number of unlimited actuators. Therefore, I^* has more rows than columns and each column has exactly one 1 in it. It is clear that I^*I^{*T} is a diagonal matrix of zeroes and ones and $I^{*T}I^*$ is the identity matrix.

Example: $I^* = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$, $I^* I^{*T} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $I^{*T} I^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Q from (2) is a positive-definite diagonal matrix which combines two functions: normalization and weighting. Variables must be normalized so that their units do not affect the feedback gains. For example, without normalization, whether an angle is measured in radians or degrees will greatly affect the solution because the squared error in degrees is much larger than in radians independent of units of other variables, thus putting more or less emphasis on the angle error compared to other variables. Once variables are normalized, they can be considered to be on the same scale. At this point it makes sense to give a greater weight to the variables which are more important, i.e. the ones which are required to most closely track their counterparts from the unlimited system. Thus it is clear from (2) that Q is the product of three *diagonal positive-definite* matrices

$$Q = N^T W N$$

where N and W perform the normalization and weighting respectively. N appears twice to account for the vectors pre- and postmultiplying Q. The transpose symbol is included for consistency even though the matrices are diagonal.

The objective function, J, in equation (2) is easily minimized as follows.

$$\begin{aligned} \frac{\partial J}{\partial u^*} &= (B\Delta u - BI^*I^{*T}u^*)^T Q B I^* I^{*T} = 0 \\ &\rightarrow I^*I^{*T}B^TQ(B\Delta u - BI^*I^{*T}u^*) = 0 \end{aligned}$$

Therefore,

$$\begin{aligned} I^*I^{*T}B^TQB\Delta u &= I^*I^{*T}B^TQBI^*I^{*T}u^* \\ I^{*T}I^*I^{*T}B^TQB\Delta u &= I^{*T}I^*I^{*T}B^TQBI^*I^{*T}u^* \\ I^{*T}B^TQB\Delta u &= I^{*T}B^TQBI^*I^{*T}u^* \end{aligned}$$

As long as $I^{*T}B^TQBI^*$ is full rank, it can be inverted, thus

$$(I^{*T}B^TQBI^*)^{-1}I^{*T}B^TQB\Delta u = I^{*T}u^*$$

which, using the identity property of I^*, leads to

$$u^* = I^*(I^{*T}B^TQBI^*)^{-1}I^{*T}B^TQB\Delta u \tag{3}$$

The elements of u^* from (3) are the incremental control commands which make the control contribution of the state vector of the limited system as close as possible in a least squares sense to that of the unlimited system. With enough redundancy, the performance can be made to match closely. In general, there is not enough redundancy to completely recover from a saturated actuator but the match might be close enough that the state variables are not noticeably disturbed. As redundancy is decreased, however, it becomes harder to compensate for future saturations. Since the Windup Feedback scheme is performance-based, the variables which are given a relatively high priority through Q should track their ideal counterparts as closely as possible at the expense of less important variables.

The crux of this work is the realization that the unmet demand, Δu, can be rewritten as

$$\Delta u = \sum_i e_i e_i^T \Delta u$$

where e_i is a column vector of zeroes with a 1 in the ith location. This implies that a completely new feedback matrix does not need to be computed using (3) each time an actuator saturates. Only the column corresponding to the newly limited effector should be determined and inserted into the existing Windup Feedback matrix. This allows for the smooth flow of control effort from saturated to unsaturated actuators without sudden spikes or jumps in the actuators' positions as additional control demand is redistributed. For step changes in actuator commands to occur, incremental commands must be suddenly added to or withdrawn from an effector, or changes must be made to existing feedback gains. The Windup Feedback scheme avoids these pitfalls by changing only the gains corresponding to the newly saturated actuator, leaving those corresponding to previously saturated actuators unchanged. This serves two purposes. First, it retains the linearity and continuity of the control signals by keeping all overflows from previously saturated actuators redistributed in the same proportion as before the most recent limit. Second, as long as the overdemand is caused by a continuous change in command, immediately after the instant of saturation, the new overflow amount is small, thus additional increments to the remaining unlimited actuators appear as smooth ramps rather than as sudden jumps. Incremental commands to redundant actuators will only appear as steps if the original overdemand was produced by the introduction of a step command.

As long as the redundancy exists to redistribute the unmet control effort over the remaining unlimited actuators, the integral windup problem associated with actuator saturation can be avoided. This is clear from the fact that windup is not due to the saturation itself, but to the sluggish response resulting from it. The redistribution of the control effort will prevent the sluggish response even though the demand may be too great for any single actuator to deliver.

Only after the redundancy has been exhausted, making accurate command following impossible, will the sluggish response occur. Thus the performance can be maintained up to the physical limits of the system. Therefore, even with the Windup Feedback algorithm, when using an integral term in the controller it is necessary to have a windup protection scheme because integrator windup can occur once all redundancy has been exploited. However, the type of windup protection chosen is up to the implementer as it is not an explicit part of the Windup Feedback algorithm.

Appendix I contains a derivation of a generic state space description of a closed loop system with saturated actuators and the Windup Feedback scheme in place. This information can be used to determine the stability of the closed loop system for different limit combinations.

From the preceding description, it is clear that this scheme meets the requirements listed above for windup protection algorithms: it is memoryless as shown by equation (3); controller stability is not affected since the incremental commands are added to the controller output, closed loop stability can be checked using the information in Appendix I; it provides bumpless transfer through the use of Windup Feedback gains which redistribute the overdemand smoothly and continuously to the available actuators; and it allows for graceful degradation by always yielding the designer-specified optimal solution to the given situation, even with multiple actuator commands saturated, as long as redundancy is available.

EXAMPLE

The XBQM-106 Unmanned Research Vehicle (URV) is an experimental aircraft developed to study flight control concepts [1]. A block diagram of the closed loop system is shown in figure 4. The seventh-order, five-input, seven-output, linearized split-surface model has coupled longitudinal- and lateral-directional equations of motion. The seven state variables correspond to the physical outputs of the aircraft and the five inputs correspond to individual actuator deflections:

angle of attack (α)
pitch angle (θ)
pitch rate (q)
sideslip angle (β)
roll angle (ϕ)
roll rate (p)
yaw rate (r)

right elevator (δ_{eR})
left elevator (δ_{eL})
right aileron (δ_{aR})
left aileron (δ_{aL})
rudder (δ_r)

The system's state space model is of the standard form in (1) with the parameter values shown in Appendix II for a cruise flight condition at Mach 0.118 and 1500 feet.

The output variables in this model are all measured in degrees so it is unnecessary to scale them. The output variables of interest correspond to the three commands: pitch, roll and yaw rate. The weighting matrix used here is shown in Appendix II. Its diagonal elements were determined through several tuning iterations.

All of the five actuators used in this example have the same transfer function: a second order characteristic with an overshoot of about four percent. The command to each actuator is bounded by the upper and lower values which would just cause saturation in steady state for that effector. The rudder was limited between ±20 degrees and the elevators and ailerons were limited between ±30 degrees. Thus a step change in demand at or near the edge of the range of motion might cause the actuator to bump the limit but not be driven hard against it.

In order to generate responses to demonstrate the Windup Feedback scheme, a turning maneuver was simulated by introducing simultaneous pitch and roll commands to the URV.

$$r = \begin{bmatrix} \theta_c \\ \phi_c \\ r_c \end{bmatrix} = \begin{cases} \begin{bmatrix} 4 \\ 4 \\ 0 \end{bmatrix} & 1 \le t < 3 \ \text{seconds} \\ \begin{bmatrix} -4 \\ -4 \\ 0 \end{bmatrix} & 3 \le t < 3.6 \ \text{seconds} \\ \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} & t < 1 \ \text{and} \ t \ge 3.6 \ \text{seconds} \end{cases}$$

Figure 5 displays plots of the pitch, roll, yaw rate and the five actuator positions for the unlimited system response, the limited but not compensated response, and the limited response with Windup Feedback. The aileron commands both saturated due to the step change in desired attitude at 3.0 seconds into the run. This resulted in two columns of the Windup Feedback matrix being generated based on an I^* corresponding to limited aileron commands:

$$I^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad u^* = \begin{bmatrix} 0 & 0 & -2.1510 & 0.3992 & 0 \\ 0 & 0 & 0.3992 & -2.1510 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.0415 & 0.0415 & 0 \end{bmatrix} \Delta u \qquad (4)$$

This redistribution of unmet demand caused the immediate saturation of the left elevator command, producing a situation where three of five actuator demands were at their limit. Thus another column of the Windup Feedback matrix was computed:

$$I^* = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad u^* = \begin{bmatrix} 0 & 0.9998 & -2.1510 & 0.3992 & 0 \\ 0 & 0 & 0.3992 & -2.1510 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -0.0373 & -0.0415 & 0.0415 & 0 \end{bmatrix} \Delta u \qquad (5)$$

which had the effect of using the right elevator to deliver the additional control effort required to track pitch. This fact is demonstrated clearly by the computed gain in the (1,2) position of the Windup Feedback matrix in (5) which redirects essentially 100% of the left elevator's overdemand to the right elevator command. Note that the unmet portion of the demand due to the limited ailerons is distributed among the three unsaturated actuators, but when the elevator limits, its unmet demand can only be distributed over the two remaining actuators. The inability of the uncompensated system to track the ideal yaw rate is obvious in both its slow reaction and its overshoot.

Figure 6 contains two sets of plots: one of the difference between the ideal case and the limited case with no compensation and the other of the difference between the ideal case and the limited case with Windup Feedback. The change in scale between figures 5 and 6 makes the improvement produced by the addition of Windup Feedback clear. There is negligible error in pitch, the most important system output. Roll and yaw rate errors were significantly reduced with the inclusion of Windup Feedback as compared to the saturated, uncompensated case. Deviation of roll from the ideal was about half of that in the uncompensated case while yaw rate strayed only about one third as far away without the oscillation.

CONCLUSIONS

The Windup Feedback scheme is able to compensate for actuator saturation as long as there is sufficient redundancy. A perfect match with the unlimited case is not possible if the state variable compensation is not adequate. The feedback scheme used attempts to minimize the difference between the control terms of the limited and unlimited cases, but a technique which matches outputs might produce even better results and would make the selection of the weighting matrix, Q, more intuitive as it would help to eliminate tuning. The scheme is simple enough to be computed on line in real time, and different values of Q can be used depending upon the flight conditions and maneuver. Guarantees and conditions on stability should be derived, even though the current technique of analyzing each limit combination can be used easily. The issue of compensating for a saturated actuator with one of a different bandwidth has not been addressed. Inclusion of dynamics in such a situation might be appropriate.

ACKNOWLEDGEMENTS

The authors wish to sincerely thank both Duane Mattern of NYMA, Inc., for his helpful discussions and insightful observations into the problem of integral windup protection, and Steve Watts of Pratt & Whitney for his practical suggestions. The authors also appreciate the support of Judy Budd, Glendell Nailing, and all of the others associated with the Summer High School Apprenticeship Research Program at NASA Lewis Research Center.

APPENDIX I: STABILITY ANALYSIS

In order to determine the effect the Windup Feedback gains will have on the closed loop system's stability, an analysis of the redistribution scheme is required, which can be performed using figure 3 as a guide. The set of commands which will saturate the actuators in steady state is known as u_{sat}, and any control signal which goes above its actuator's saturation value will not have its full effect felt. The objective of the Windup Feedback algorithm is to compensate for any control signals which surpass their limit by rechannelling the unmet demand through redundant unsaturated actuators. The control vector which drives the actuators, u_L, must have no component greater than the corresponding element of u_{sat} so that it does not force the actuator against its stop. This is ensured by making u_L the output of a limiter which truncates any excessive components at their saturation value. Any difference, Δu, between what goes into and what comes out of the limiter is multiplied by the Windup Feedback gain matrix, K, to produce an incremental control vector, u^*. These incremental commands are added to the unbounded output of the control mixer K_M, u_{un}, to form a new control vector, u^+, which serves as the input to the limiter.

$$\begin{aligned} u^{+} = u_{un} + K\Delta u = u_{un} + Ku^{+} - Ku_{L} &\Rightarrow (I - K)u^{+} = u_{un} - Ku_{L} \\ &\Rightarrow u^{+} = (I - K)^{-1}(u_{un} - Ku_{L}) \end{aligned}$$

$$\begin{aligned} u_{L} &= I^{*}I^{*T}u^{+} + (I - I^{*}I^{*T})u_{sat} \\ &= I^{*}I^{*T}(I - K)^{-1}(u_{un} - Ku_{L}) + (I - I^{*}I^{*T})u_{sat} \end{aligned}$$

$$\Rightarrow (I + I^{*}I^{*T}(I - K)^{-1}K)u_{L} = I^{*}I^{*T}(I - K)^{-1}u_{un} + (I - I^{*}I^{*T})u_{sat}$$

$$\begin{aligned} \Rightarrow u_{L} &= (I + I^{*}I^{*T}(I - K)^{-1}K)^{-1}I^{*}I^{*T}(I - K)^{-1}u_{un} + (I + I^{*}I^{*T}(I - K)^{-1}K)^{-1}(I - I^{*}I^{*T})u_{sat} \\ &= Lu_{un} + Mu_{sat} \end{aligned}$$

THEOREM: $(I - K)$ is always invertible.

PROOF: Columns of K are computed such that the (i,i)-element is always zero since that position corresponds to the saturated actuator. Without loss of generality, assume actuators saturate from *1*st to *n-1*st. This way all computed gains will lie below the main diagonal. The last actuator, whether it saturates or not, causes no change in the final column since there are no redundant actuators left. Thus, when computed from left to right, $I - K$ will be lower triangular with ones on the main diagonal ⇒ it is full rank. If saturations come in a different order, rows and columns of K are just interchanged, not affecting rank. ■

THEOREM: $(I + I^{*} I^{*T}(I - K)^{-1} K)$ is always invertible.

PROOF: Assuming as above that $(I - K)$ is lower triangular, $(I - K)^{-1}$ is also lower triangular. Combined with the fact that $(I - K)^{-1} (I - K) = I \Rightarrow (I - K)^{-1} K = (I - K)^{-1} - I$ it is clear that the nonzero elements of $(I - K)^{-1} K$ all lie below the main diagonal. Since $I^{*}I^{*T}$ is a diagonal matrix of zeroes and ones, it will select individual rows of $(I - K)^{-1}$ K so that every nonzero element of $I^{*} I^{*T}(I - K)^{-1}K$ is below the main diagonal. Thus, $(I + I^{*} I^{*T}(I - K)^{-1}K)$ will be lower triangular with ones on the main diagonal ⇒ it is full rank and therefore invertible. ■

Referring to figure 3, it can be seen that in this general form there can be four dynamical systems, not all of which may be present in any particular implementation: $G_C(s)$, the controller in the forward loop; $G_a(s)$, the actuators; $G(s)$, the plant under control; and $H(s)$, the controller in the feedback loop. Since the actuators, $G_a(s)$, and the plant, $G(s)$, are mechanical systems, we will assume that they are strictly proper, thus the D matrices associated with their state space realizations are zero. The controllers have no such assumptions placed on them and indeed most likely will have a proportional component in their response. Therefore, the state equations for the closed loop system are as follows.

$$\begin{aligned} \dot{x}_a &= A_a x_a + B_a u_L \\ \dot{x} &= A\,x + B\,u_a \\ \dot{x}_H &= A_H x_H + B_H y \\ \dot{x}_c &= A_c x_c + B_c e \end{aligned}$$

where

$$
\begin{aligned}
u_a &= C_a x_a \\
y &= Cx \\
e &= r - C_H x_H - D_H C x \\
u_L &= L C_c x_c + L D_c e - M u_{sat} \\
&= L C_c x_c + L D_c r - L D_c C_H x_H - L D_c D_H C x - M u_{sat}
\end{aligned}
$$

thus, in state space form

$$
\begin{bmatrix} \dot{x}_a \\ \dot{x}_c \\ \dot{x}_H \\ \dot{x} \end{bmatrix} = \begin{bmatrix} A_a & B_a L C_c & -B_a L D_c C_H & -B_a L D_c D_H C \\ 0 & A_c & B_c C_H & B_c D_H C \\ 0 & 0 & A_H & B_H C \\ B C_a & 0 & 0 & A \end{bmatrix} \begin{bmatrix} x_a \\ x_c \\ x_H \\ x \end{bmatrix} + \begin{bmatrix} B_a L D_c \\ B_c \\ 0 \\ 0 \end{bmatrix} r + \begin{bmatrix} -B_a M \\ 0 \\ 0 \\ 0 \end{bmatrix} u_{sat} \qquad (6)
$$

We will concern ourselves with the eigenvalues of the system matrix above. The matrix L in the case with no saturations is merely the identity matrix. Therefore, in the unlimited case we may assume that the controller was designed to stabilize the system, resulting in a negative semi-definite matrix (no eigenvalues greater than zero).

In order to test the stability of a closed loop system with Windup Feedback, simply substitute the appropriate submatrices into the above system matrix and determine the eigenvalues. In the situations where there is only one dynamic controller, either in the forward path or in the feedback path but not both, the system matrix above must have a block row and block column deleted to account for the zeroth-order characteristic. For instance, in a system with unity feedback, A_H, B_H, and C_H do not exist while D_H equals the identity matrix of appropriate dimension. It is important to understand that A_H does not exist rather than equal 0 since a matrix of zeroes implies a set of pure integrators. Similarly, if no dynamic controller exists in the forward path, A_c, B_c, and C_c, are zero-dimensional while D_c is a transformation of some type, usually a control mixer.

APPENDIX II: EXAMPLE

The state space representation of each of the subsystems in the closed loop Unmanned Research Vehicle system from the example has the form

$$
\begin{aligned}
\dot{x} &= Ax + Bu \\
y &= Cx + Du
\end{aligned}
$$

The open loop URV has the following parameters:

$$
A = \begin{bmatrix} -2.632 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -39.63 & 0 & -3.382 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.7399 & 0.2442 & 0.00096 & -0.9849 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -17.6 & 0 & -8.94 & 3.16 \\ 0 & 0 & 0 & 22.09 & 0 & 0.137 & -2.05 \end{bmatrix}
$$

$$B = \begin{bmatrix} 0.0073 & 0.0073 & -0.01 & -0.01 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0.5118 & 0.5118 & -0.0697 & -0.0697 & 0 \\ 0 & 0 & 0.00055 & -0.00055 & -0.00159 \\ 0 & 0 & 0 & 0 & 0 \\ -0.2489 & 0.2489 & 0.7356 & -0.7356 & -0.0685 \\ 0.0132 & -0.0132 & -0.0464 & 0.0464 & 0.1809 \end{bmatrix} \times \begin{bmatrix} 2.815 & 0 & 0 & 0 & 0 \\ 0 & 2.815 & 0 & 0 & 0 \\ 0 & 0 & 3.6 & 0 & 0 \\ 0 & 0 & 0 & 3.6 & 0 \\ 0 & 0 & 0 & 0 & 11.0 \end{bmatrix}$$

$$C = \frac{180}{\pi} \times I_7$$

In the equation for B above, the matrix on the left represents how the forces and moments due to the control surfaces affect each state variable. The matrix on the right corresponds to the linkage ratio associated with each actuator and therefore represents the amplification of each control signal.

In this example the five actuators models are all the same, resulting in a block diagonal state space representation of

$$A_a = diag\left(\begin{bmatrix} -12.7 & -162.71 \\ 1.0 & -12.7 \end{bmatrix}, \begin{bmatrix} -12.7 & -162.71 \\ 1.0 & -12.7 \end{bmatrix}, \begin{bmatrix} -12.7 & -162.71 \\ 1.0 & -12.7 \end{bmatrix}, \begin{bmatrix} -12.7 & -162.71 \\ 1.0 & -12.7 \end{bmatrix}, \begin{bmatrix} -12.7 & -162.71 \\ 1.0 & -12.7 \end{bmatrix} \right)$$

$$B_a = diag\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right)$$

$$C_a = diag\left(\begin{bmatrix} 0 & 324 \end{bmatrix}, \begin{bmatrix} 0 & 324 \end{bmatrix}, \begin{bmatrix} 0 & 324 \end{bmatrix}, \begin{bmatrix} 0 & 324 \end{bmatrix}, \begin{bmatrix} 0 & 324 \end{bmatrix} \right)$$

The controller in the forward loop is zeroth-order and consists of only the mixer matrix

$$D_c = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The dynamic controller in the feedback path may be represented in state space as

$$A_H = \begin{bmatrix} -8 & 0 & 0 \\ 0 & 0 & 4 \\ 0 & -1 & -4 \end{bmatrix}$$

$$B_H = \begin{bmatrix} 0 & 0 & -2.6667 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.1188 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.0297 \end{bmatrix}$$

$$C_H = \begin{bmatrix} 0.014 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$D_H = \begin{bmatrix} 0 & 0.0541 & 0.0187 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0556 & 0.0063 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.0297 \end{bmatrix}$$

The weighting matrix, Q, is shown below. Each diagonal element provides the relative emphasis to place upon the corresponding state variable and thus it plays a significant role in determining the Windup Feedback gains. The elements of Q were selected, through several tuning iterations, to minimize the differences between the ideal and the compensated responses in roll, pitch, and yaw rate for the example turning maneuver. The best set of responses and thus the corresponding Q matrix was determined through visual comparison of plotted data.

$$Q = \begin{bmatrix} 10{,}000{,}000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 750{,}000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1000 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1000 \end{bmatrix}$$

In order to illustrate the mechanism at work in this example, numerical values will be used to demonstrate the state variable compensation. Recall that the two aileron commands saturated simultaneously which resulted in a nonzero Δu. Thus, the unmet portion of the command, Δu, corresponded to the unmet state variable contribution, $B \Delta u$. If, for example,

$$\Delta u = \begin{bmatrix} 0 \\ 0 \\ -1 \\ 1 \\ 0 \end{bmatrix}, \qquad B \Delta u = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -0.0040 \\ 0 \\ -5.2963 \\ 0.3341 \end{bmatrix} \tag{7}$$

Ideally, using the Windup Feedback gains in (4) in an attempt to compensate for this discrepancy, close matching is achieved, thus

$$u^* = \begin{bmatrix} 2.5502 \\ -2.5502 \\ 0 \\ 0 \\ 0.0830 \end{bmatrix}, \quad Bu^* = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -0.0015 \\ 0 \\ -4.1994 \\ 0.3548 \end{bmatrix}$$

However, in the example, the increment to the left elevator command was enough to cause it to saturate, creating a third nonzero component of Δu. If, for instance, at a particular point in time this new component equals -1, the command to the left elevator is one unit beyond the lower limit of what that actuator can deliver. This means that the sum of u and the computed u^* will be truncated at the limit value, reducing the effective u^* and again disrupting the state variable compensation. Thus there is a large deviation between the required compensation and the available compensation using just the original two columns of the Windup Feedback matrix.

$$u^* = \begin{bmatrix} 2.5502 \\ -2.5502 \\ 0 \\ 0 \\ 0.0830 \end{bmatrix} - \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2.5502 \\ -1.5502 \\ 0 \\ 0 \\ 0.0830 \end{bmatrix}, \quad Bu^* = \begin{bmatrix} 0.0205 \\ 0 \\ 1.4407 \\ -0.0015 \\ 0 \\ -3.4987 \\ 0.3176 \end{bmatrix}$$

Notice that u^* is reduced by the amount of the additional Δu component. Feeding back the overdemand portion of the left elevator command through the new gains in (5) will produce some additional command increments

$$\Delta u = \begin{bmatrix} 0 \\ -1 \\ -1 \\ 1 \\ 0 \end{bmatrix}, \quad u^* = \begin{bmatrix} 2.5502 \\ -2.5502 \\ 0 \\ 0 \\ 0.0830 \end{bmatrix} - \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -0.9998 \\ 0 \\ 0 \\ 0 \\ 0.0373 \end{bmatrix} = \begin{bmatrix} 1.5504 \\ -1.5502 \\ 0 \\ 0 \\ 0.1204 \end{bmatrix}, \quad Bu^* = \begin{bmatrix} 0 \\ 0 \\ 0.0003 \\ -0.0021 \\ 0 \\ -3.0793 \\ 0.3547 \end{bmatrix} \tag{8}$$

Thus even with three of the five actuator commands limited, the Windup Feedback scheme nearly returns the state variables to their desired values, i.e., the components due to the incremental commands, Bu^*, in (8) nearly match the components due to the overdemand, $B\Delta u$, in (7). The exception is in the state variable corresponding to roll rate which had a relatively low priority through the weighting matrix Q. Keep in mind that when there are multiple saturations transiently, as in this example, the overdemand is constantly changing and is thus redistributed in continuously varying proportions among the different actuator groupings. Consequently, the accuracy of the state variable compensation will change with time as a varying amount of overdemand is redirected to less effective redundant actuator combinations. In this example, all of the state variables are given in radians or radians per second so direct comparison of their magnitudes is reasonable. The table below lists the eigenvalues, determined using equation (6), for the three cases in this example: unlimited, both ailerons saturated, both ailerons and the left elevator saturated. In all three cases the closed loop system remains stable.

Table of Eigenvalues for Unlimited and Limited Systems

unlimited	ailerons saturated	ailerons and left elevator saturated
-0.3154 + 0.0000j	-0.3154 + 0.0000j	-0.0887 + 0.0000j
-1.2245 + 1.0757j	-1.1061 + 0.9776j	-1.0145 + 0.6385j
-1.2245 - 1.0757j	-1.1061 - 0.9776j	-1.0145 - 0.6385j
-2.6298 + 0.0000j	-2.4271 + 0.0000j	-2.1602 + 0.0000j
-4.4131 + 3.6903j	-4.4680 + 3.6257j	-4.4132 + 3.7420j
-4.4131 - 3.6903j	-4.4680 - 3.6257j	-4.4132 - 3.7420j
-4.7279 + 7.0123j	-4.7279 + 7.0123j	-4.9812 + 6.9037j
-4.7279 - 7.0123j	-4.7279 - 7.0123j	-4.9812 - 6.9037j
-9.6675 + 0.0000j	-9.6675 + 0.0000j	-9.5553 + 0.0000j
-11.4745 + 0.0000j	-11.3187 + 0.0000j	-11.4154 + 0.0000j
-8.5260 +11.1786j	-8.5731 +11.1486j	-8.5296 +11.0844j
-8.5260 -11.1786j	-8.5731 -11.1486j	-8.5296 -11.0844j
-9.9877 +10.9354j	-9.9877 +10.9354j	-9.7236 +10.8223j
-9.9877 -10.9354j	-9.9877 -10.9354j	-9.7236 -10.8223j
-12.0492 +12.0933j	-12.2447 +12.2911j	-12.7000 +12.7558j
-12.0492 -12.0933j	-12.2447 -12.2911j	-12.7000 -12.7558j
-12.7000 +12.7558j	-12.7000 +12.7558j	-12.7000 +12.7558j
-12.7000 -12.7558j	-12.7000 -12.7558j	-12.7000 -12.7558j
-12.7000 +12.7558j	-12.7000 +12.7558j	-12.7000 +12.7558j
-12.7000 -12.7558j	-12.7000 -12.7558j	-12.7000 -12.7558j

REFERENCES

1. K. Rattan, in *Proc. IEEE 1985 NAECON*, pp. 560-569.
2. A. K. Caglayan, S. M. Allen, K. Wehmuller, in *Proc. IEEE 1988 NAECON*, pp. 520-529.
3. J. S. Litt, preprint number 19, in *AHS Rotary Wing Propulsion Specialists' Meeting* (1990).
4. B. Noble, *Applied Linear Algebra*, (Prentice-Hall, Inc., Englewood Cliffs, NJ, 1969), pp. 142-145.
5. S. Adibibhatla et al., *STOVL Controls Technology*, NASA CR 195361 , Final Report (1993).
6. P. J. Campo, M. Morari, C. N. Nett, in *Proc. American Control Conference* (1989), pp. 1706-1711.
7. R. Hanus, *Journal A*, **21**, 15 (1980).

8. D. Mattern, AIAA-93-3812, in *Proc. AIAA Guidance, Navigation, and Control Conference* (1993).

9. Watts, Stephen R., and Garg, Sanjay, "An Optimized Integrator Windup Protection Technique Applied to a Turbofan Engine Control," NASA TM 107035.

10. Marcopoli, Vincent R., and Phillips, Stephen M., "Analysis and Synthesis Tools for a Class of Actuator Limited Multi-input Multi-output Control Systems," NASA CR to be published.

11. Boyd, S.; El Ghaoui, L.; Feron, E.; and Balakrishnan, V., *Linear Matrix Inequalities in System and Control Theory*, SIAM, 1994.

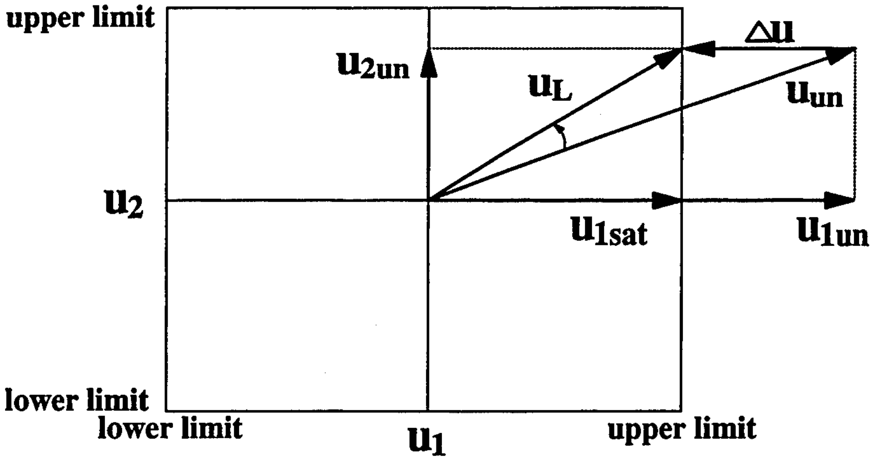


Figure 1.—Example of Directionality Problem where a saturated actuator changes the direction of the control vector, which is constrained to lie within the box.

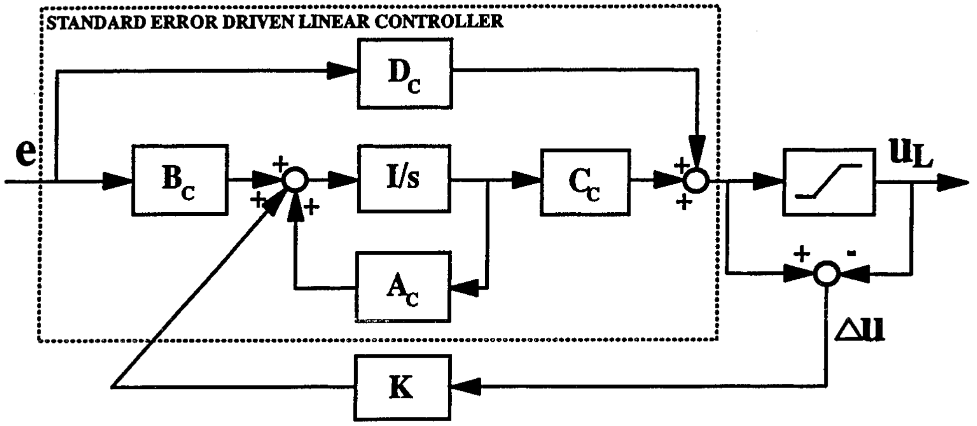


Figure 2.—Block diagram of a controller with the typical form of Integral Windup Protection implemented.

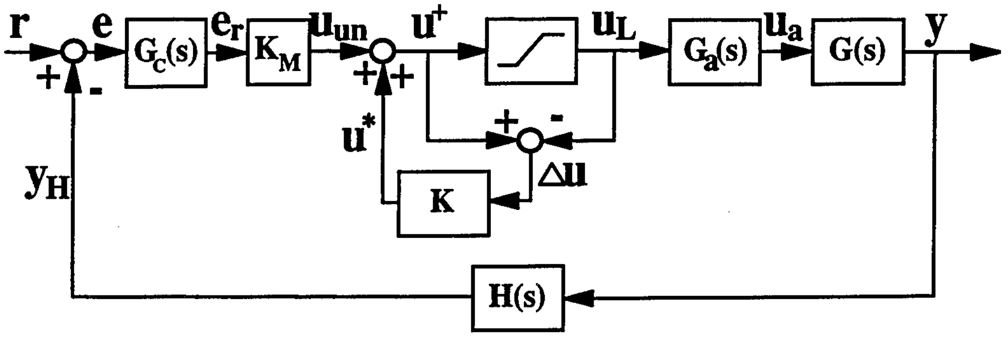


Figure 3.—Block diagram of a system with the Windup Feedback scheme implemented, showing all variables.

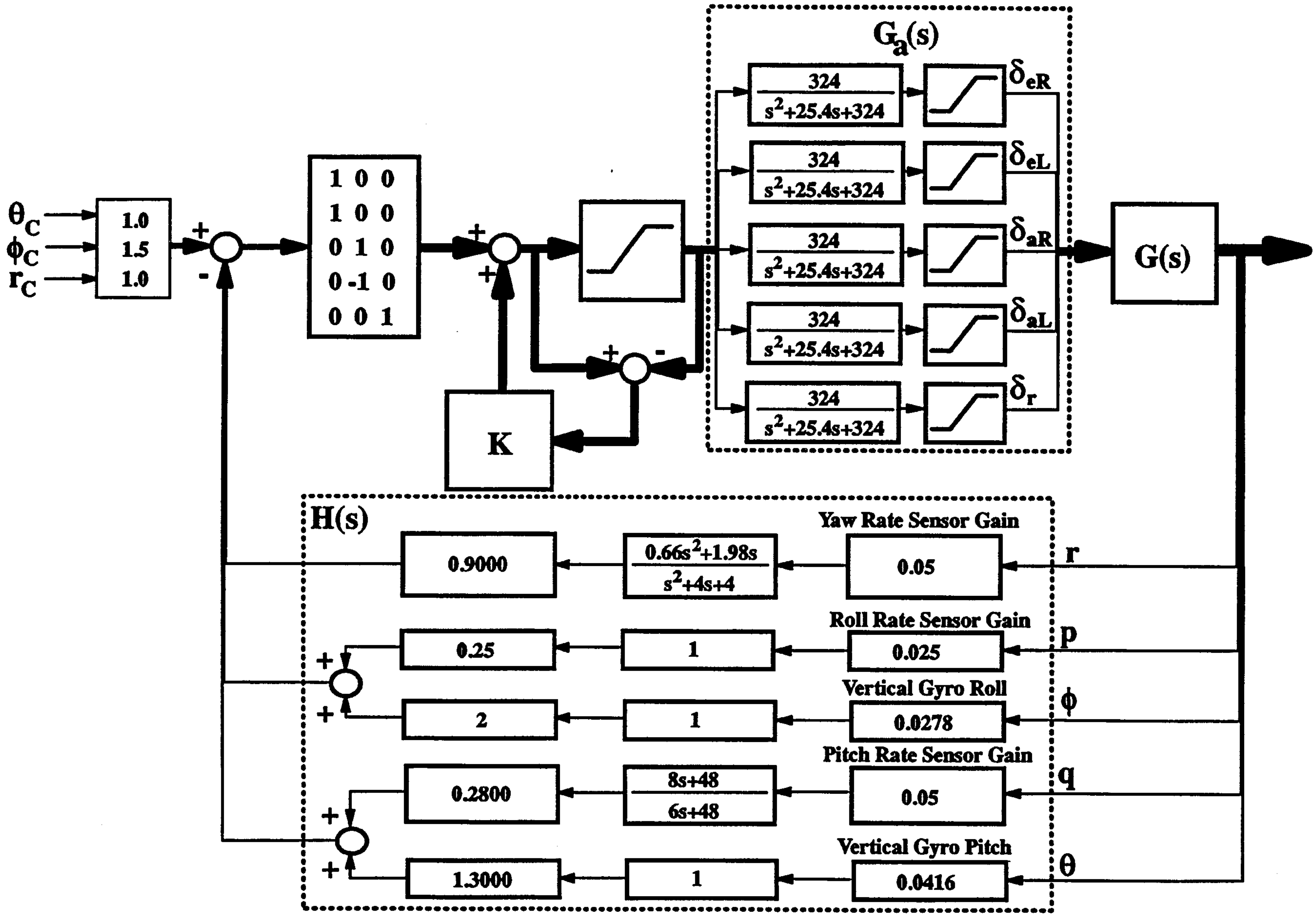
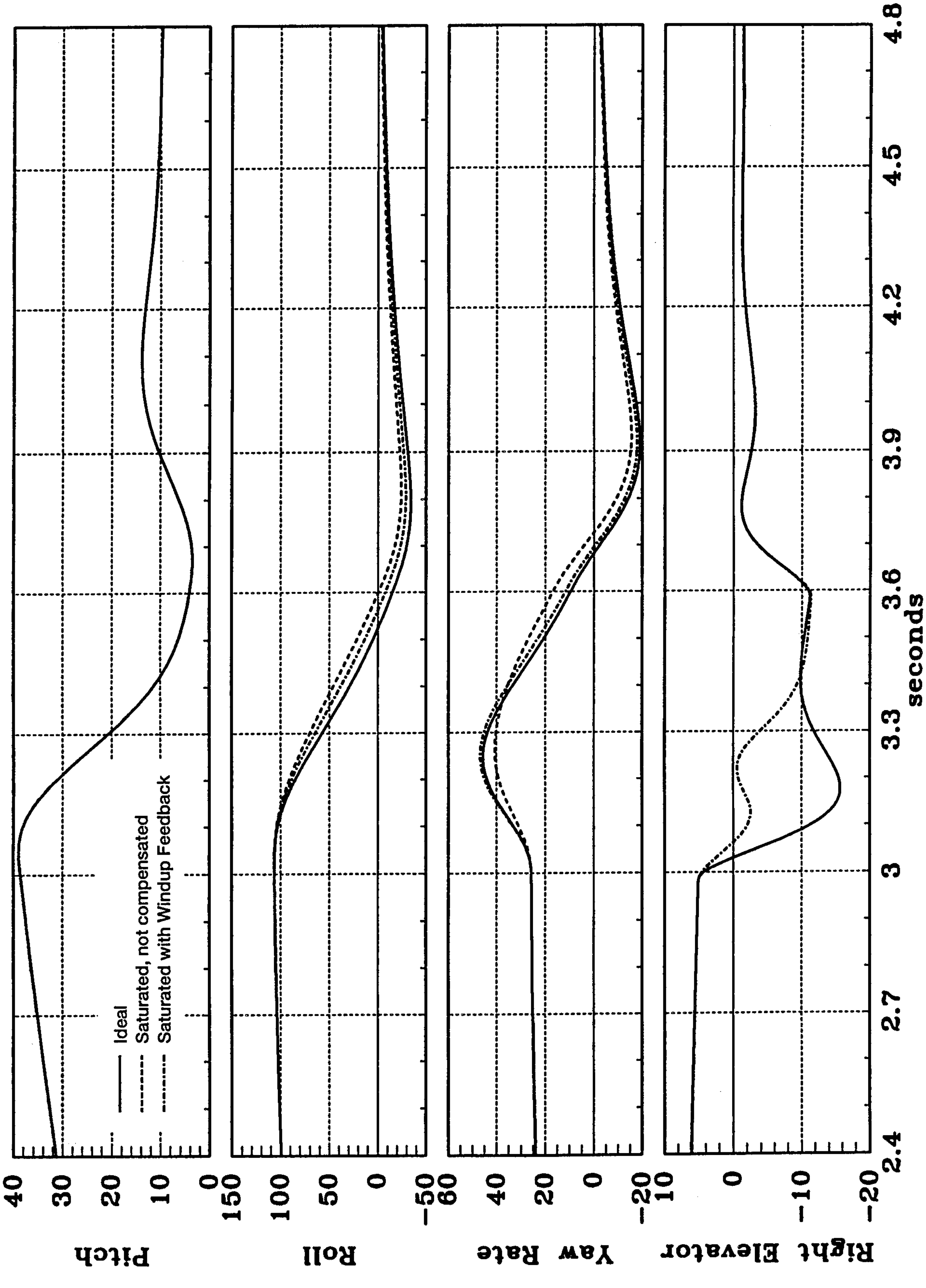


Figure 4.—Block diagram of URV example.


Ideal
Saturated, not compensated
Saturated with Windup Feedback
Pitch
40
30
20
10
0
Roll
150
100
50
0
−50
Yaw Rate
60
40
20
0
−20
Right Elevator
10
0
−10
−20
2.4
2.7
3
3.3
3.6
3.9
4.2
4.5
4.8
seconds

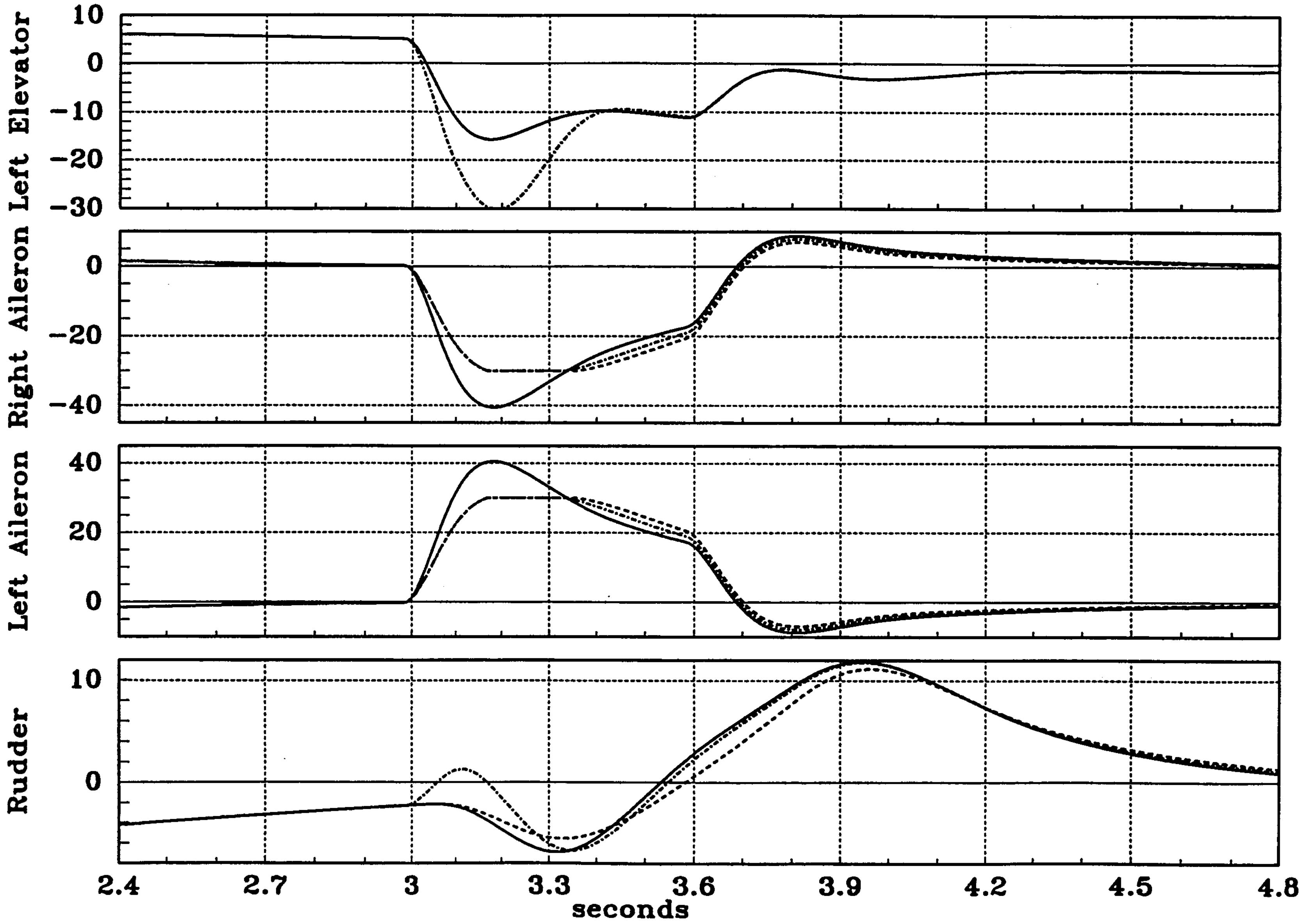
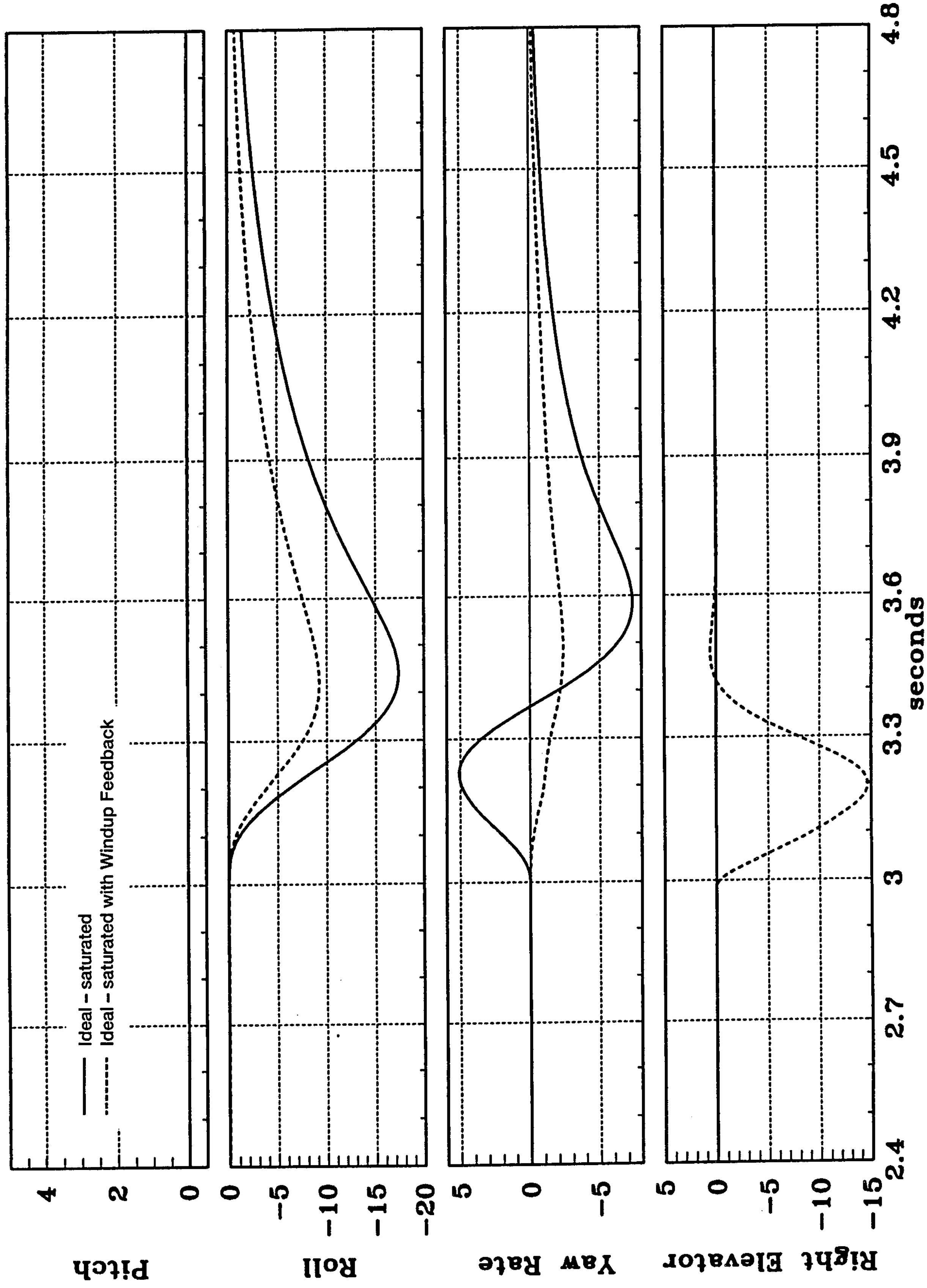


Figure 5.—Full scale plots of URV example.


Ideal – saturated
Ideal – saturated with Windup Feedback
Pitch
4
2
0
Roll
0
−5
−10
−15
−20
Yaw Rate
5
0
−5
Right Elevator
5
0
−5
−10
−15
2.4
2.7
3
3.3
3.6
3.9
4.2
4.5
4.8
seconds

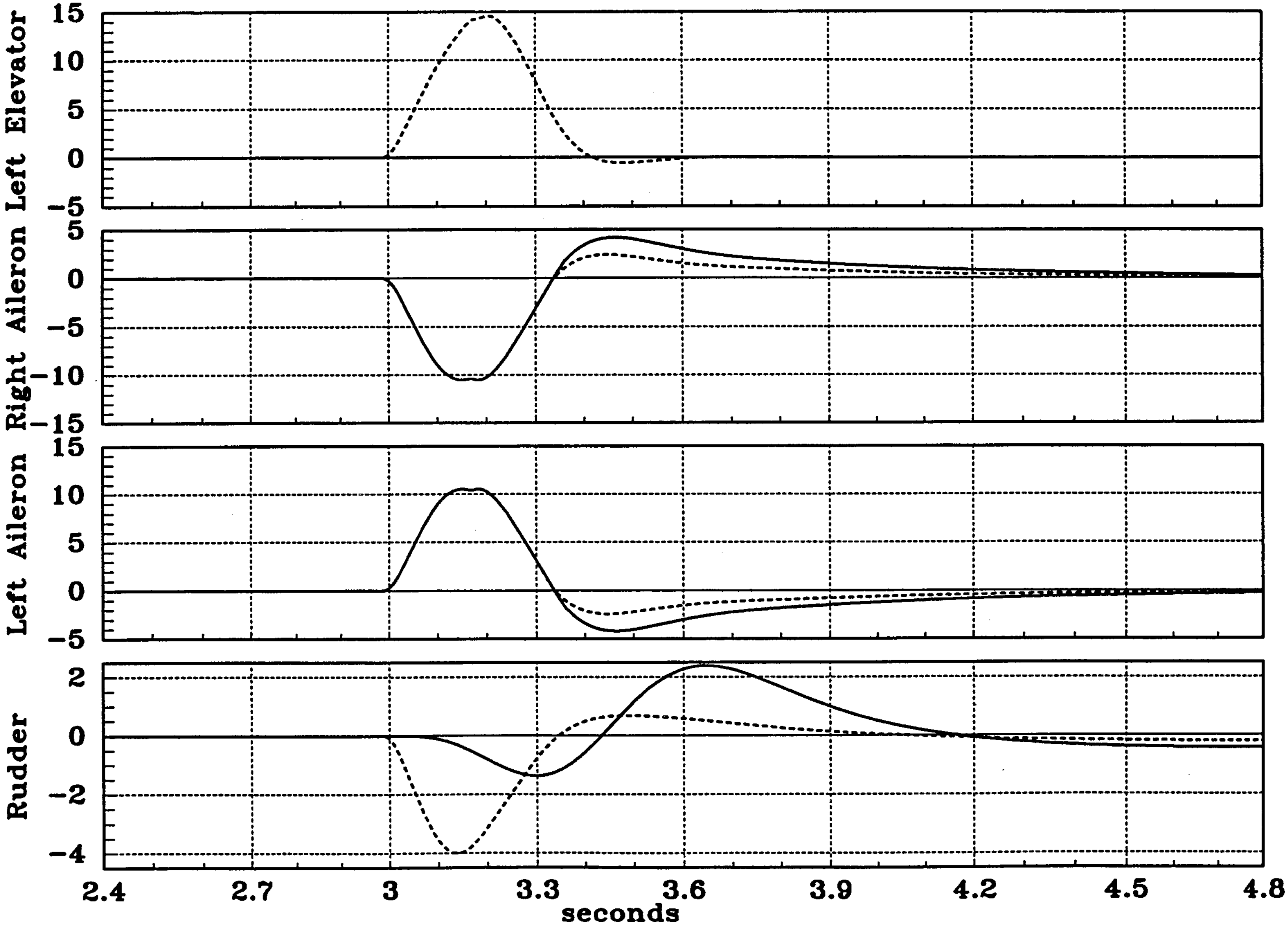


Figure 6.—Differential plots of the URV example.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1995	Technical Memorandum

4. TITLE AND SUBTITLE

A Method for Exploiting Redundancy to Accommodate Actuator Limits in Multivariable Systems

5. FUNDING NUMBERS

WU-505-62-0L
1L161102AH45

6. AUTHOR(S)

Jonathan Litt and Greg Roulette

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

NASA Lewis Research Center
Cleveland, Ohio 44135-3191
and
Vehicle Propulsion Directorate
U.S. Army Research Laboratory
Cleveland, Ohio 44135-3191

8. PERFORMING ORGANIZATION REPORT NUMBER

E-9460

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Washington, D.C. 20546-0001
and
U.S. Army Research Laboratory
Adelphi, Maryland 20783-1145

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

NASA TM-106859
ARL-TR-906

11. SUPPLEMENTARY NOTES

Prepared for the 19th Science Conference sponsored by the U.S. Army, Orlando, Florida, June 20-24, 1994. Jonathan Litt, Vehicle Propulsion Directorate, U.S. Army Research Laboratory, NASA Lewis Research Center; and Greg Roulette, Shaker Heights High School, Shaker Heights, Ohio and Summer High School Apprenticeship Research Program participant at Lewis Research Center. Responsible person, Jonathan Litt, organization code 2530, (216) 433-3748.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified - Unlimited
Subject Category 63

This publication is available from the NASA Center for Aerospace Information, (301) 621-0390.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

This paper introduces a new method for accommodating actuator saturation in a multivariable system with actuator redundancy. Actuator saturation can cause significant deterioration in control system performance because unmet demand may result in sluggish transients and oscillations in response to setpoint changes. To help compensate for this problem, a technique has been developed which takes advantage of redundancy in multivariable systems to redistribute the unmet control demand over the remaining useful effectors. This method is not a redesign procedure, rather it modifies commands to the unlimited effectors to compensate for those which are limited, thereby exploiting the built-in redundancy. The original commands are modified by the increments due to unmet demand, but when a saturated effector comes off its limit, the incremental commands disappear and the original unmodified controller remains intact. This scheme provides a smooth transition between saturated and unsaturated modes as it divides up the unmet requirement over any available actuators. This way, if there is sufficiently redundant control authority, performance can be maintained.

14. SUBJECT TERMS

Multivariable control; Integrator windup; Control systems

15. NUMBER OF PAGES

23

16. PRICE CODE

A03

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	