ISSN 0280–5316
ISRN LUTFD2/TFRT--5562--SE

Control of Unstable Systems with Input and Rate Saturations

Michelangelo Calabresi

Department of Automatic Control
Lund Institute of Technology
August 1996

Department of Automatic Control
Lund Institute of Technology
Box 118
S-221 00 Lund Sweden

Document name
MASTER THESIS

Date of issue
August 1996

Document Number
ISRN LUTFD2/TFRT--5562--SE

Author(s)
Michelangelo Calabresi

Supervisor
K. J. Åström

Sponsoring organisation
ERASMUS

Title and subtitle

Control of Unstable Systems with Input and Rate Saturations

Abstract

Input saturation and rate limitations are important, particularly for unstable systems. This work explores three control problems that illustrate this. The impact of a saturation and a rate limiter on a plant controlled with a PID controller is one problem. Anti-windup techniques provide good solutions for systems with saturation but unfortunately they do not work so well for systems with rate limitations. Modifications of the conventional anti-windup schemes are presented to overcome these problems. A second problem deals with the phase lag induced by rate limiters and several filters are presented and compared showing different kind of performances for each one. Analysis shows that no filter is superior in all respects. The third problem deals with unstable multivariable systems. It is based on a linear model of the pitch motion of the fighter aircraft JAS-39 Gripen. Modern fighter planes have highly unstable dynamics and the limited forces produced by the aerodynamic surfaces must match diverging actions caused by the intrinsic aerodynamic instability. Several techniques are used to design a controller that improves the handling qualities of the plane while avoiding that the pilot looses control of the plane. Hybrid control has been implemented to provide smooth transition between a normal flight condition and a critical one and appears to be a promising way to deal with this kind of problems. Hybrid control can also overcome the nonlinear behaviour of a real airplane at high angles of attack.

Key words

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280–5316

ISBN

Language
English

Number of pages
122

Recipient's notes

Security classification

CONTENTS

1. Introduction

Linear systems have several desirable properties that ease enormously their analysis and make it possible to know all the properties of the system through a limited number of tests.
This is why it is often convenient to approximate, whenever possible, a real system with a linear model; in most cases it is possible to define an operating boundary within which the system may be considered linear, and linear controllers are designed to keep the system within these boundaries.
Given for example the frequency response of a linear system to an unitary signal, we are able to predict the response to any signal, no matter the magnitude it will have.
This is obviously not true with physical systems, we cannot expect for instance that feeding an electrical engine with an increasing input current will lead to a forever proportionally increasing output.
In case of general actuators we will have concrete limits of the output that they can provide; additionally, such output may not be linearly dependent from the input and sometimes two equal inputs may lead to two different outputs, according to the 'history' of the inputs.
Additionally there can be physical limits to the rate at which we can change the system's input due to mechanical limits of the structure. For example fast inputs may produce resounding frequencies into elastic systems resulting into undesirable vibrations, or simply an actuator may not be able to instantly provide a certain level of output but must be given some time to reach it, according to its physical construction, its functioning principle and the nature of the phenomena used to generate the output.
During the following chapters attention will be focused on two kind of non linear elements, saturations and rate limiters, which are used to simulate many common features of real systems. This work explores three control problems related to this.
The impact of a saturation and a rate limiter on a plant controlled with a PID controller is one problem. Anti-windup techniques provide good solutions for systems with saturation but unfortunately they may not work so well for systems with rate limitations. The problem will be approached from several directions. When the rate limits are not tight, some simple changes to the anti-windup scheme provide a good answer. When instead the system input has very strong rate limitations, a finite time is required to move the signal between different output levels. An extra derivative effect, before the controller, compensates for these delays and makes it very easy to tune the controller.
A second problem deals with the phase lag induced by rate limiters. A rate limiter is able to correctly reproduce signals whose derivative does not exceed a certain value. If instead the derivative of the signal exceeds the rate limit, the output will be affected by an amplitude loss and a phase lag. (see next picture)
Most systems have no more than 50-60 degrees of phase margin. Unstable airplanes are stabilized with control systems which rarely give more than 45 degrees of phase margin.
A rate limiter may induce, in the worst case, a phase lag of 90 degrees, which is enough to make the system unstable. Rate limiters were considered the main cause of many PIO[1] accidents.
To overcome these effects, several filters may be designed in the attempt of reducing the phase lag. In this work three main kind of filters are presented and analyzed and a comparison is made to evaluate their different performances.

[1] PIO : Pilot Induced Oscillations.

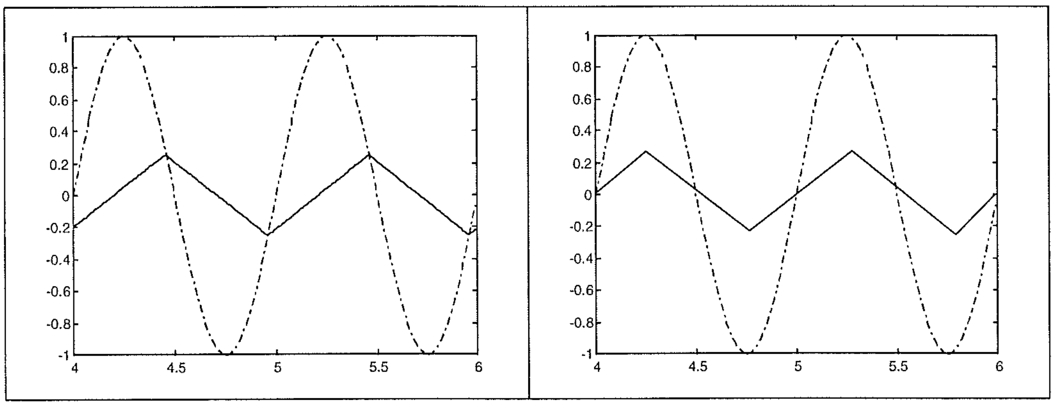


Figure 1

On the left, the response of a rate limiter (solid line) to a signal with absolute values of the derivative which exceed the rate limit. To the right an ideal response of the same rate limiter, with zero phase lag.

Different kind of filters have different properties and it is not possible to find a filter that combines zero phase lag with a good response to all kind of signals. Even if it is possible to constantly improve the performance of different filters, it will become clear that the goal of the perfect compensation can never be reached due to the nature of the problem.

In the third part, a linear model of the pitch motion of the JAS-39 Gripen is studied.

Modern fighter airplanes have highly unstable aerodynamic designs in order to maximize performance and efficiency. The limited forces produced by the aerodynamic surfaces must match diverging actions caused by the intrinsic aerodynamic instability. Several techniques are used to design a controller that improves the handling qualities of the plane while avoiding that the pilot looses control of the plane.

Usually modern airplanes with digital control systems are equipped with an 'auto-recovery' system which assumes the total control of the plane when some state variable exceeds certain critical limits.

In this work an attempt has been made to design a controller whose dynamic does not allow the pilot to exceed the operating boundaries given by the control saturations while at the same time leaving always authority to the pilot's inputs. Unfortunately such specifications on a controller are in conflict with the requirement for good handling qualities of the aircraft.

Hybrid control has been implemented to provide smooth transition between a normal flight condition and a critical one and appears to be a promising way to deal with this kind of problems.

Hybrid control can also overcome the non-linear behavior of a real airplane at high angles of attack.

The presence of input rate saturations requires predictive actions. An input signal takes a while to change from a value to another and during this time, the unstable dynamic will move further away from the initial position. This must be accounted for when designing an emergency recovery system.

Generally an additional fixed safety margin, calculated with a worst-case approximation, is always added due to the presence of rate limiters [2].

In this work the possibility of applying dynamic boundaries is explored. A dynamic boundary is constantly calculated according to the current state of the input signal. Dynamic boundaries allow full use of the operating range.

In general, harmful effects caused by rate limiters may always be counteracted or at least significantly reduced, through the use of more sophisticated controller algorithms.

In a digital controller this means having a more complex software, which is therefore more likely to present problems of reliability.
In mission-critical applications computer reliability and software reliability are a major issue and therefore this aspect must be taken into account when deciding upon a control strategy.

1.1 Saturation

In most cases a physical system will not be able to provide an unlimited output, but instead there will be a maximum value to such output. The most easy way to simulate this is to have an element that behaves like a normal unitary gain (i.e. does not modify the signal) provided that the input keeps within certain given boundaries. If the boundaries are exceeded then the block provides the closest value possible, keeping it constant until the input returns into the acceptable range.

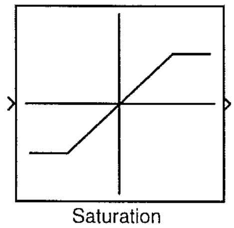


Figure 2
Common symbol for a Saturation block

A saturation element is therefore defined by two values : the minimum value *m* and the maximum value *M* and given an input *u* to the element the output will follow the law :

Input	Output
u < m	m
m < u < M	u
u > M	M

such behavior can be also represented trough a graph :

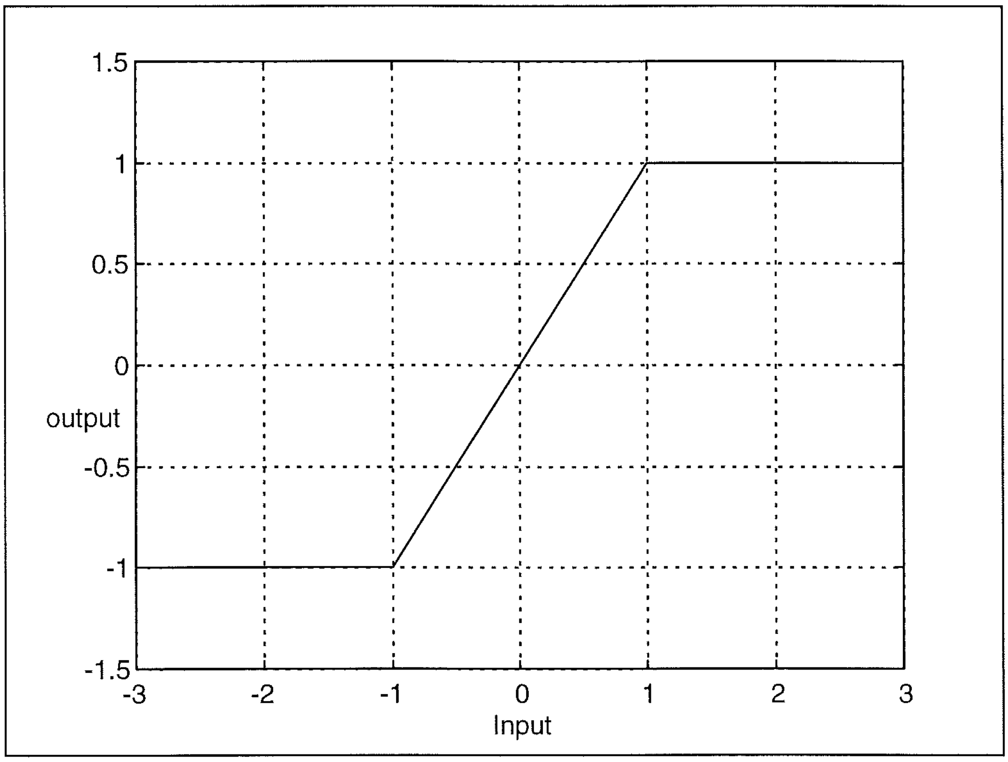


Figure 3
Input-Output relation of a typical saturation block

Of course there are several undesirable characteristics associated with a saturation ; the most obvious one is the fact that the controller cannot use arbitrarily high inputs and therefore there is a concrete limit to the performance of the system.

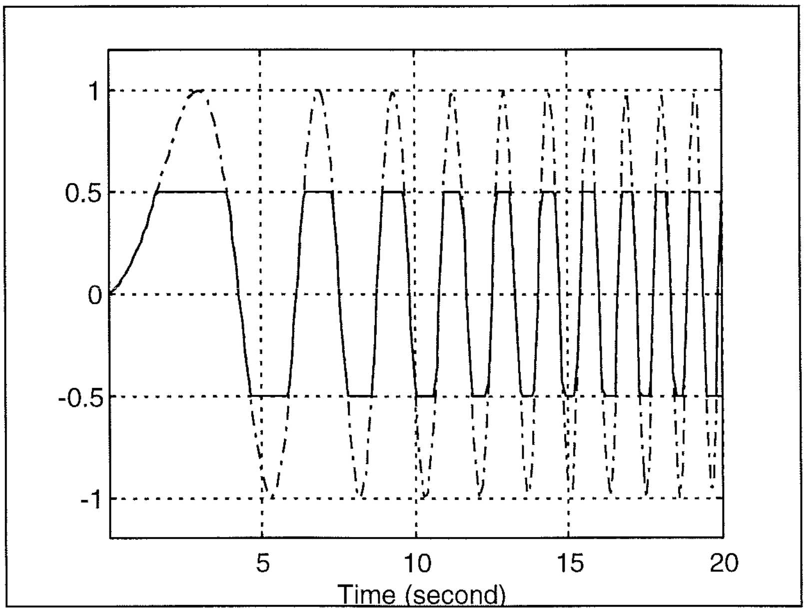


Figure 4
Time response of a saturation block to a signal with increasing frequency

In a physical system we can improve the controller to increase performance and increase efficiency, but at a certain point we will reach a limit when no concrete improvements are

reachable simply changing the logic of the controller, instead changes in the physical system such as more powerful actuators will be required. Those changes will reflect in the model into less tight saturation limits.
When designing a controller it is important to notice that when the limits of the saturation are exceeded, the control loop is broken and there is no more relation between changes in the input and changes in the output, since the output will remain invariably constant regardless of any small changes in the input, provided that the input remains out of the saturation's linear range.
This may lead to problems in controllers that have for instance integrating components since the controller may overestimate the ability of the system to follow the given input and when registering a prolonged error due to the slower dynamic imposed by the saturation uses this error to constantly increase the integral, to levels that are not useful and leading the system to large overshoots. This phenomenon will be better described in chapter 2.
When a saturation is associated with an unstable linear system, more critical risks arise. In an unstable linear system large control efforts are required to keep and recover the system from positions that are far away to the equilibrium condition and this means that there are some limits beyond which it is not possible anymore, due to the saturation limit, to provide converging inputs and the system diverges indefinitely. This other aspect of saturation-related problems will be covered in the 4th and 5th chapters.

1.2 Rate limiter

A rate limiter is a device that limits the derivative of the signal. In other terms a rate limiter passes to the output port the input signal *u* if and only if the condition :

$$\left|\frac{du}{dt}\right| < r$$

is satisfied, where r is the rate limit that characterizes the non-linearity.
If instead the input's changing rate is greater, then the output moves towards the input value at its maximum rate, until it manages to join again the signal.
Of course general rate limit may admit the existence of a 'falling' rate limiter that is different than the 'raising' one, meaning that the maximum rate at which the output may increase can be different than the maximum rate at which the output may decrease.

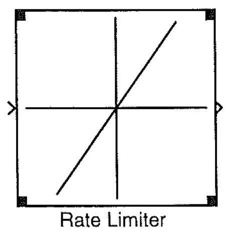


Figure 5
Common symbol for a Rate-Limiter Block

In the case of the rate limiter there is not a fixed relation between the input and output values, since it is depending on the history of the signal, therefore it is not possible to represent the link between the input and the output with a simple graph. This also leads to

the fact that the graphic representation of a rate limiter is less representative of the effective behavior of the block.
Rate limiters themselves do not limit the maximum value of the signal, but put limits on the rate of change of the signal itself ; such limiters usually are associated with some problems related with the designing of controllers.

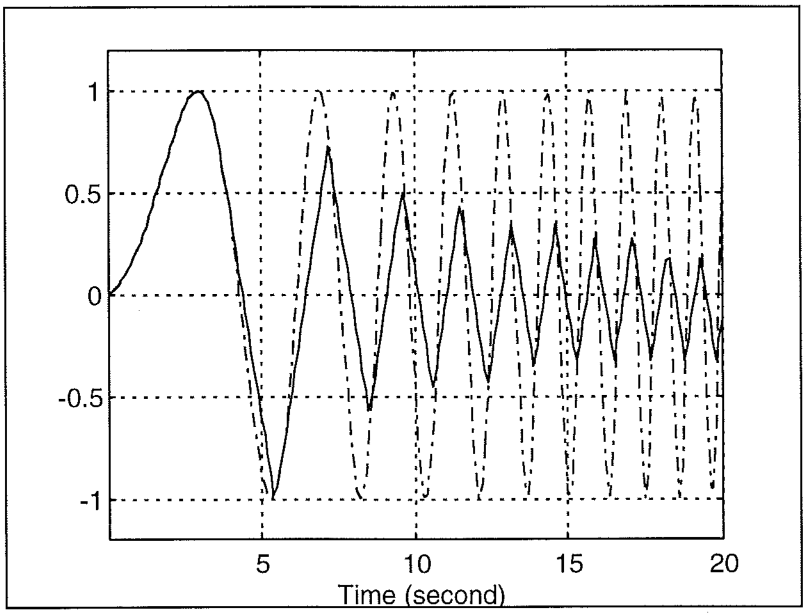


Figure 6
Typical time response for a rate limiter to a sinusoidal input of increasing frequency

Generally a rate limiter delays the way some signals change and this generically result in a loss of phase margin. For some frequencies and for some gains this phase loss may reach 90 degrees, a value high enough to treat the stability of a marginally stable system. Chapter 3 will better analyze the problem and will present several filters aimed at reducing this phase lag.
Under normal conditions controllers designed without taking into account the rate limits will also suffer from the finite time needed to reverse or just change the input signal, and this will delay the time a controller needs to lead the system to the desired set-point ; eventually long overshoots are associated with rate limits for the same reasons.

1.3 Combining a rate limiter with a saturation

Often (if not always) the rate limit alone does make little meaning in a physical system, since as already said a saturation is almost always present, representing real systems' ability to manage signals up to a certain size.
Therefore in most model rate limits will be combined with a saturation into a single non linear function that will include both the effects.
It is worthwhile to notice that in a block diagram representation of the system, the combination of a rate limiter and a saturation gives different results according to the order of placement. In other words, placing along the same signal stream a rate limiter followed by a saturation may in some case lead to a different time response that the case in which the rate limiter is placed after the saturation.
As an example we will consider the two systems as follows :

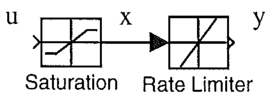


Figure 7
Combining a saturation with a rate limiter

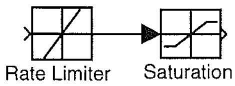


Figure 8
Combining a rate limiter with a saturation

If we analyze the time response to a slow and large sinusoidal input, we observe that the first setting results into a easily predictable output : the resulting signal loses track of the input due to the excess of speed and then stops at the saturation limit, waiting for the signal to get back down to reasonable values. When this occurs the signal starts following the input in the opposite direction. (see next figure)

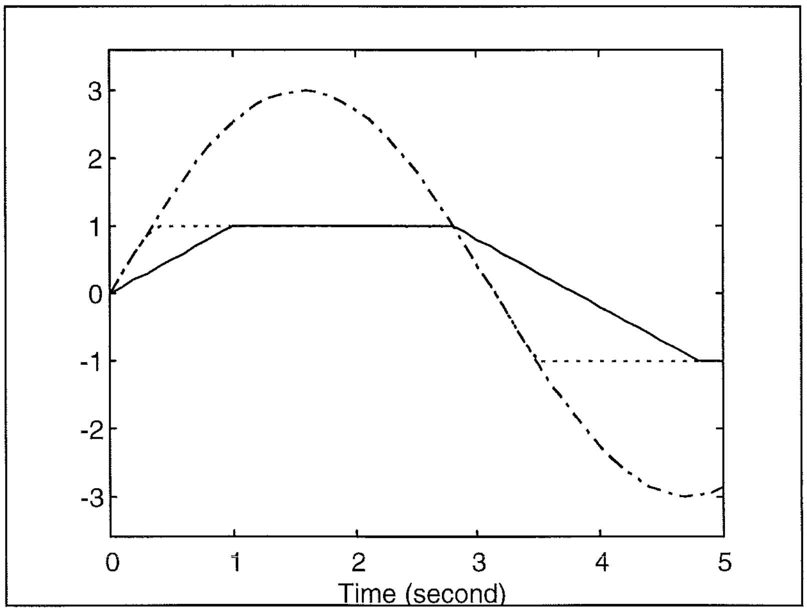


Figure 9
Time response of a saturation-rate limiter combination to a low-frequency large-amplitude sinusoidal signal.

If instead we feed the same input signal to the second block arrangement, the results are somewhat disappointing : after the saturation limit has been reached, the intermediate signal x continues to increase and this causes a delay when it comes to reverse the output direction.

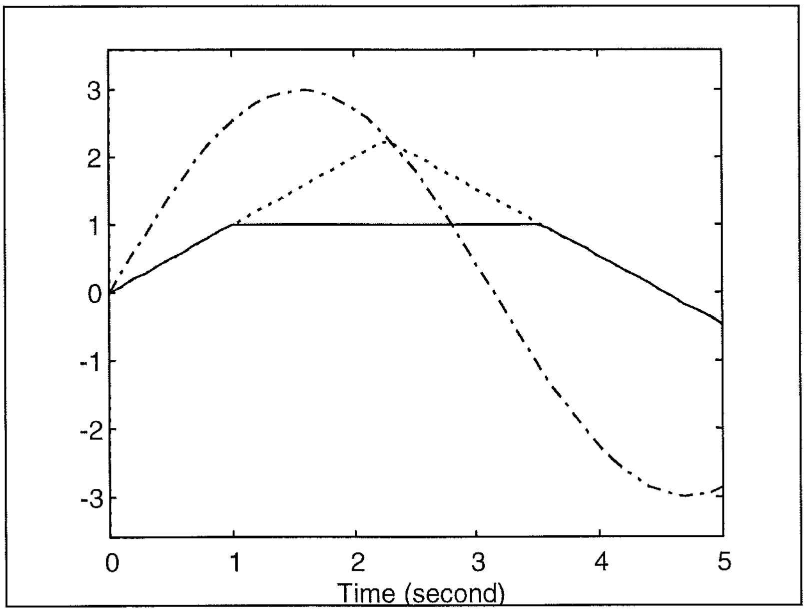


Figure 10
Time response of a rate limiter-saturation combination to a low-frequency large-amplitude sinusoidal signal.

The two arrangements are clearly leading to different responses and the basic difference between the two is an additional phase lag caused by the combination rate limiter-saturation that is instead reduced by the opposite combination.
This can be clearly be seen by the following two figures, showing a steady-state time response for both combinations and with the same parameters of the above examples :

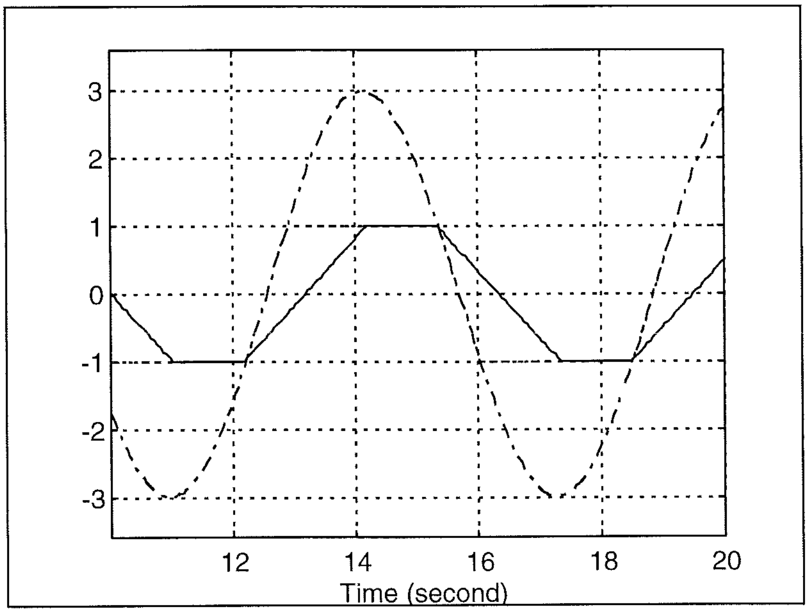


Figure 11
Steady state response to a sinusoidal signal of a saturation-rate limiter combination

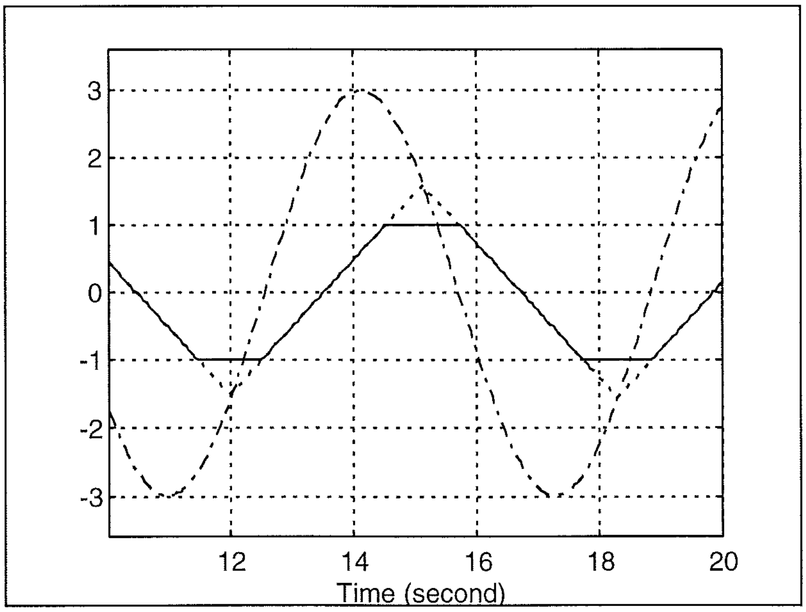


Figure 12
Steady state response to a sinusoidal signal of a rate limiter-saturation combination

In the second case the phase lag is sensibly greater and is the same phase lag than the one that would be caused by a simple rate limiter without saturation.
Instead, a saturation properly placed before the rate limiter reduces the phase lag.
It has been already said that a potentially dangerous disadvantage of the rate limiter is a loss of phase margin and therefore is straightforward to notice that when combining the blocks the saturation should always be preceding the rate limiter.
Unless the contrary is specified, n the rest of this paper this assumption will always be done.

2. PID Control

2.1 Introduction

In this chapter, simple plants will be regulated with PID controllers, then saturations and rate-limitations will be added and possible modifications to the controllers will be explored to improve the response and minimize the overshoots caused by the non-linearities.
Anti-windup circuits provide good compensation to the harmful effects caused by a saturation in the control loop ; several other solutions are presented to overcome the presence of rate limits that in some cases can not be simply compensated tuning the PID controller or changing the parameters in the anti-windup blocks.
The anti-windup systems all include logic and switching. It is a non trivial task to analyze the stability at systems that combine logic and control; for this reason the schemes suggested have been explored extensively by simulation.
Only a few of the simulations are given in the report.

2.2 Simple PID system

In this basic system, the plant will be represented by a simple integration and the controller is a standard PID. The input will be an initial step function with a value of 6 and then, after 20 seconds, another step function with a value of -4 will be added, giving a final input of 2.
A scope is added to the integral action to track its value during the evolution of the system ; the value of the integral action will usually represented on the graphs with a dash-dotted line.

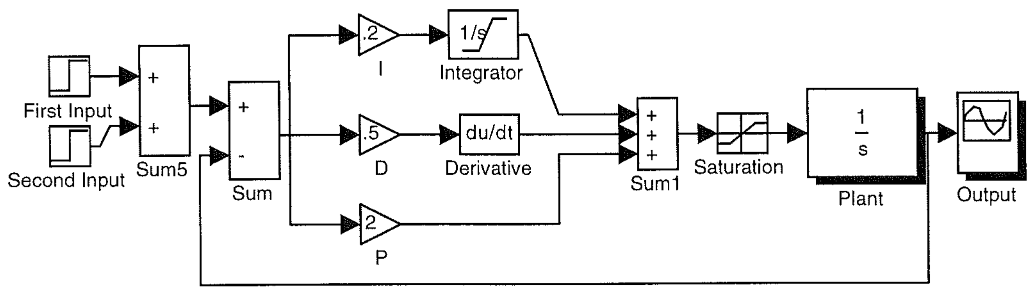


Figure 1
Simple PID system

At the beginning we will keep the saturation limit quite large (100) so that the system can evolve without feeling substantial decrease in performance from it (see next figure).
With such a simple plant is rather easy to tune the controller correctly ; integral and derivative actions are not required to achieve an optimal response, the only reason why integral action is always included is for means of error rejection.

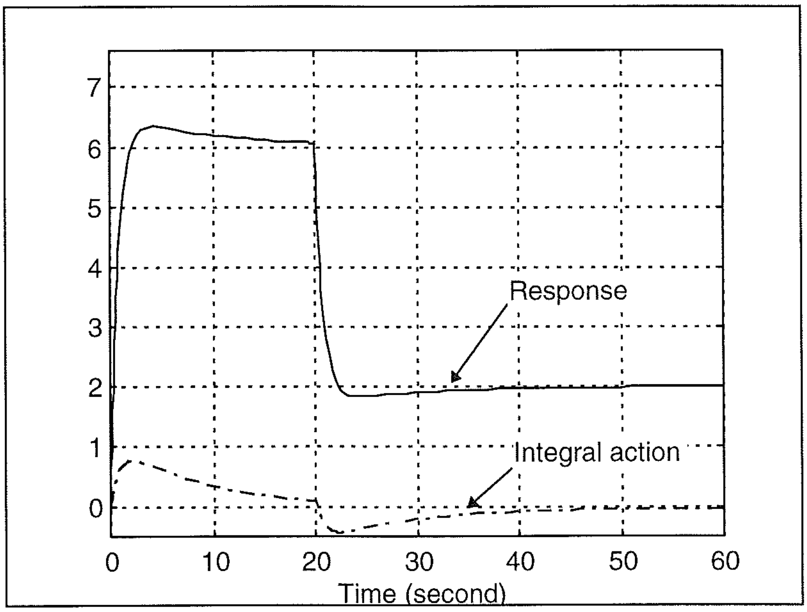


Figure 2
Response to a test signal made up by two step functions added together

Now, imposing a very narrow saturation limit of +/- 1 we can clearly observe the system's overshoot caused by the integral action, visible in the picture, as usual, as a dash-dotted line.

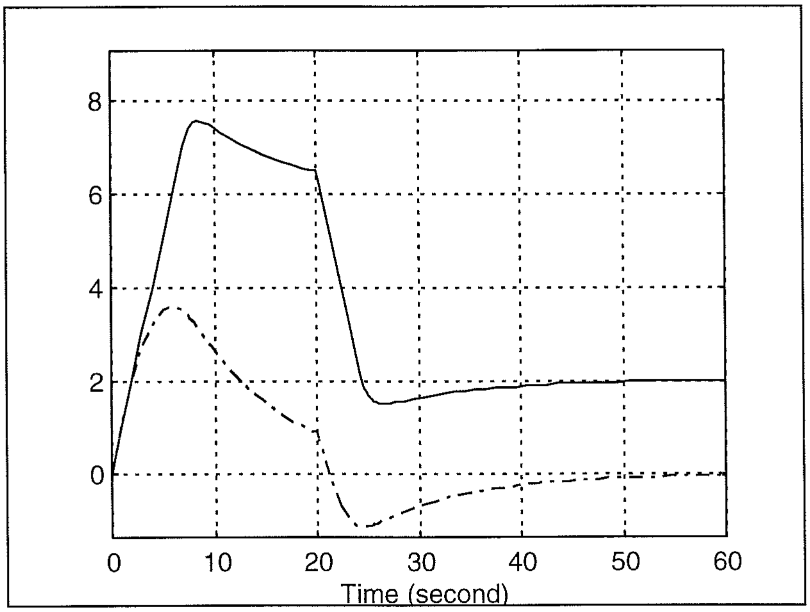


Figure 3
The presence of a saturation causes overshoots around the set points

Introducing a saturation limit of just +/- 0.5 effect is definitely impairing the ability of the system to track our inputs, as shown in the next figure :

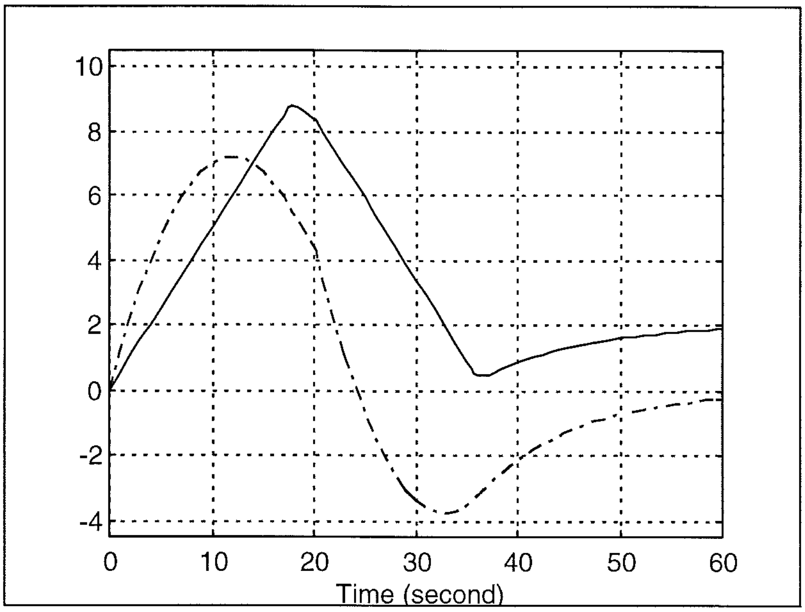


Figure 4
A tight saturation impairs completely the tracking ability of the controller

2.3 Anti-windup

A classical solution to this kind of problems is the introduction of an anti-windup feedback to the integral part of the controller. The anti-windup takes effect whenever the actuator is saturated and prevents the integral term to continue to charge while the saturation is breaking the control loop.

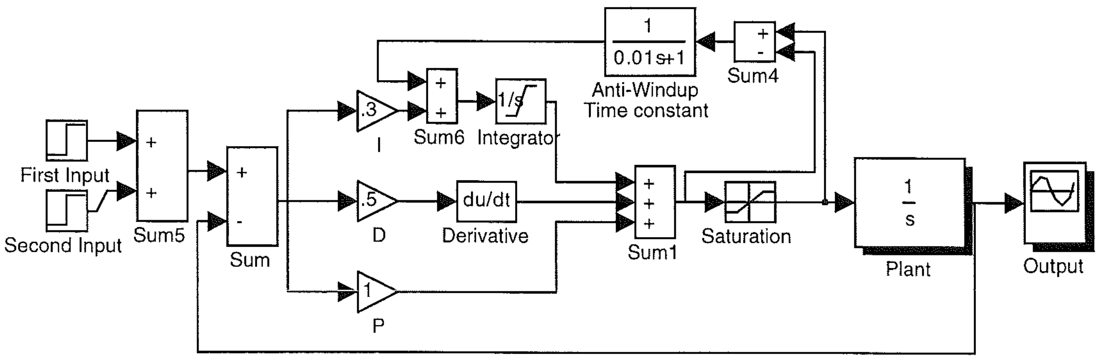


Figure 5
Anti-windup

With an anti-windup the same simulation shows a much better response ; now the integral action is kept from overloading and the overshoot is minimal.

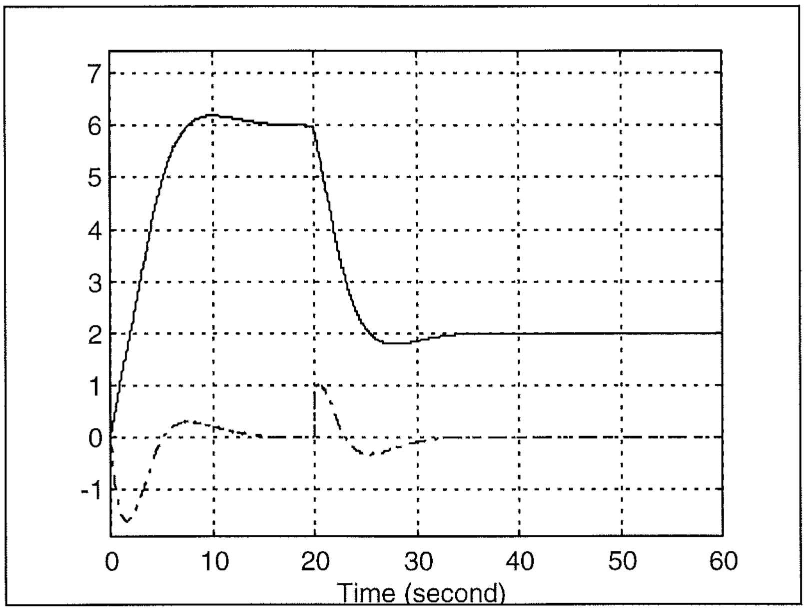


Figure 6
Improved response through the use of an anti windup circuit

2.4 Integral-discharging logic

A less sophisticated way to solve the problem is to introduce a circuit that discharges the integral term when its action tends to bring the system away from the set point.
The circuit compares the sign of the error with the sign of the integral action and if the result is negative, the integral term is quickly brought to zero.
The overshooting only depends from how much quickly the integral buffer is discharged.

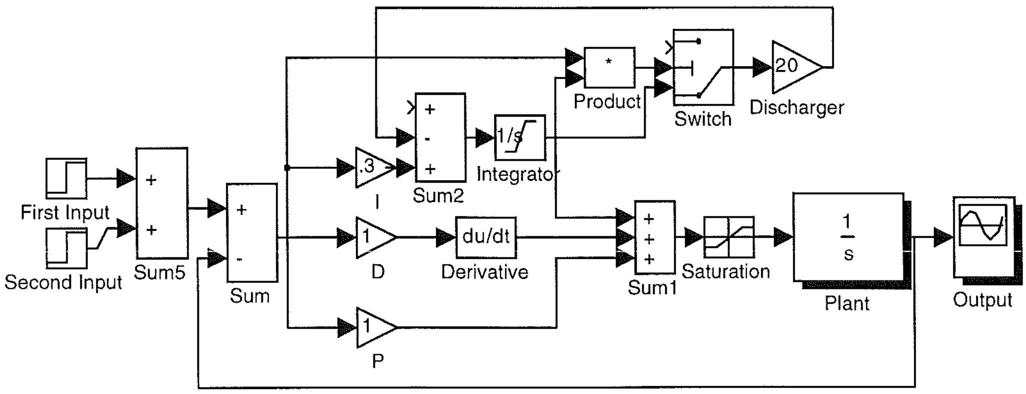


Figure 7
PID controller with integral discharger

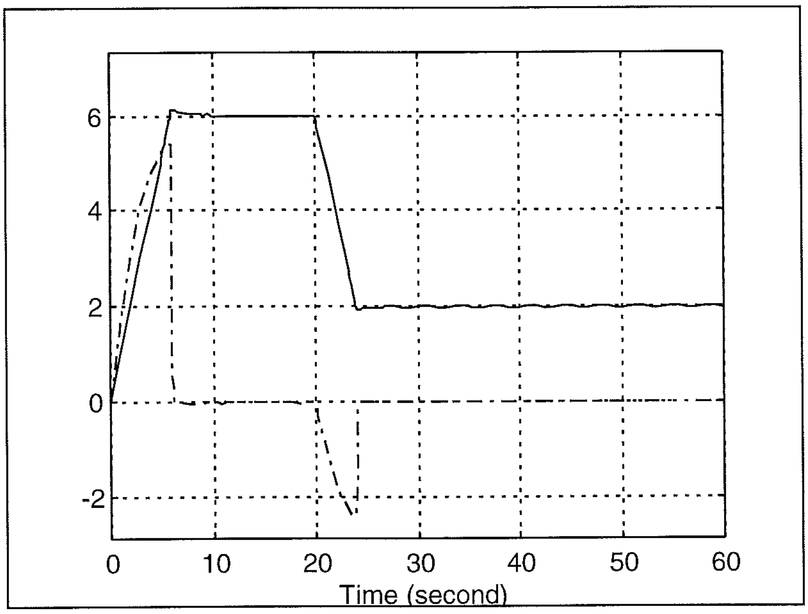


Figure 8
Response of a PID controller with integral discharger

Evidently a strong limit of this controller is the need of an integration in the plant, otherwise when the system reaches the set point, it will still need a non-zero integral action, but the controller will think that this action is diverging and will discharge it, causing the system to diverge back from the set-point until the integral has charged again.
In case of a plant without an integral action it will be necessary to modify the controller so that the steady-state value of the integral is constantly calculated and fed to the discharger, every time the input changes.
Still, the system will be vulnerable and will have a very bad response to disturbances that cause an offset of the system from its set point and further changes are needed to introduce a smarter logic.
To show this limit we will now add a disturbance in the form of a negative step function added after 10 seconds between the controller and the plant, while at 30 seconds the input will be set to 6 with a step function as input

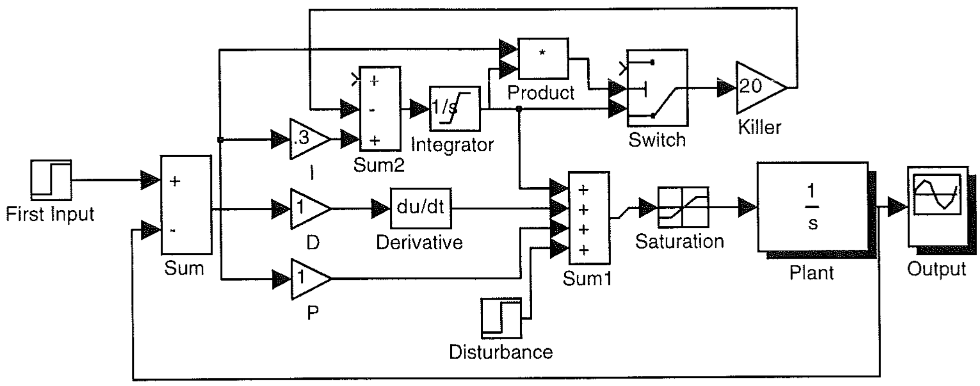


Figure 9
A disturbance before the plant is used to test the ability of the PID with integral discharge to reject non-modeled disturbances

The result is a highly irregular pattern due to the fact that the integral part is discharged immediately after reaching the set point much below the level needed to keep the correct level of output.

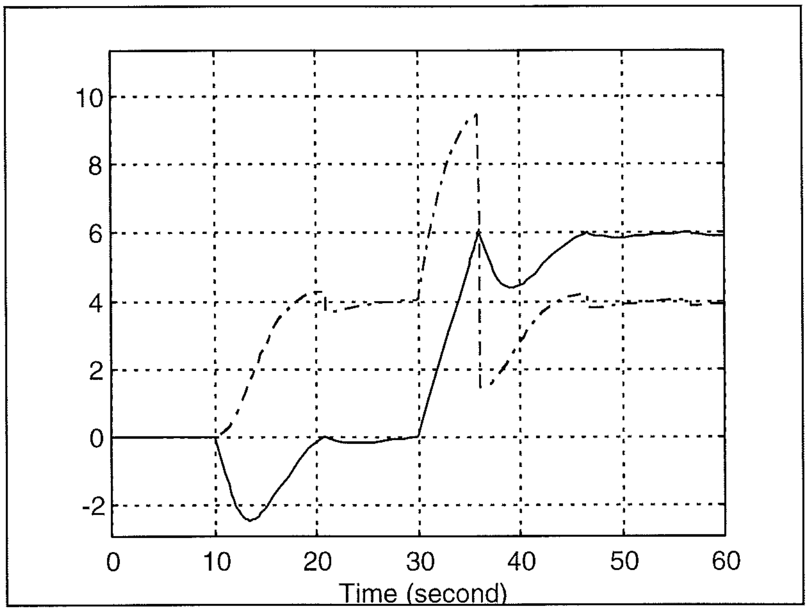


Figure 10
The disturbances causes severe problems to the controller since the integral is discharged even when useful, due to the disturbance that requires a non-zero value for the integral at steady-state.

This can be corrected with a much more complicated version of the integral discharger shown in the next figure where a cascade made up of three different switches is activated only in two cases : the first case occur when the integral action is positive, the error is negative and the derivative if the error is negative as well ; the other case occur when these conditions are satisfied with all opposite signs.
As usual, the fulfillment of these conditions causes the progressive discharge of the integral buffer, while every other combination has no effect.
The effect of this new circuit is to provide much more tight restrictions before discharging the integral. Now there are three conditions that must be satisfied before an integral can be defined harmful and therefore there are less chances that an useful integral is brought to zero.

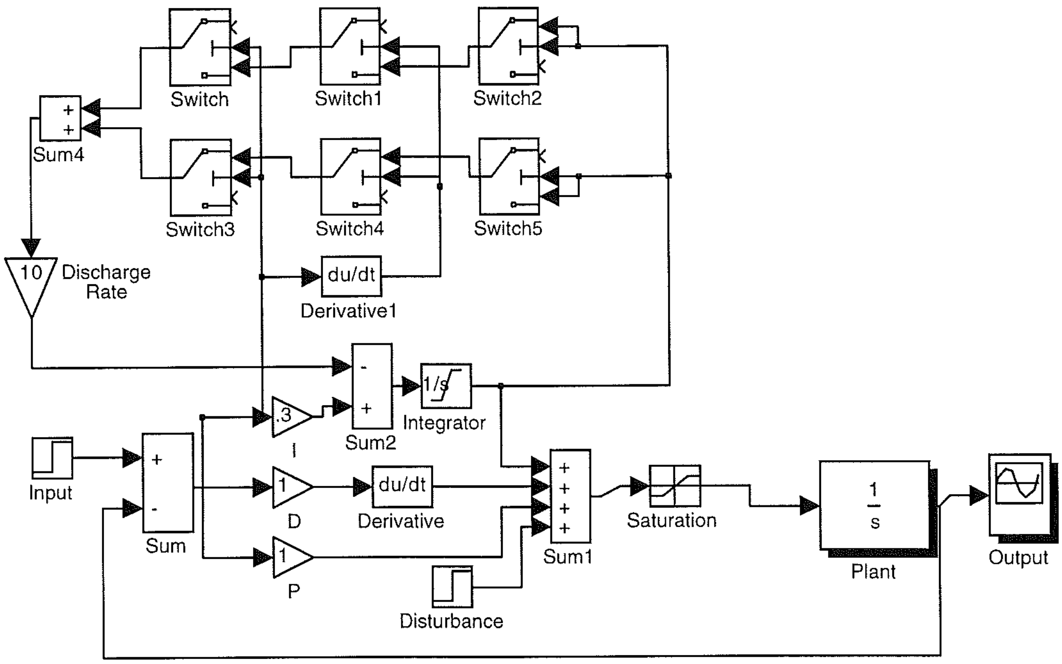


Figure 11
Improved logic for the integral discharged adds more strict requirements needed to discharge the integral buffer.

What follows is the response to the same kind of disturbance :

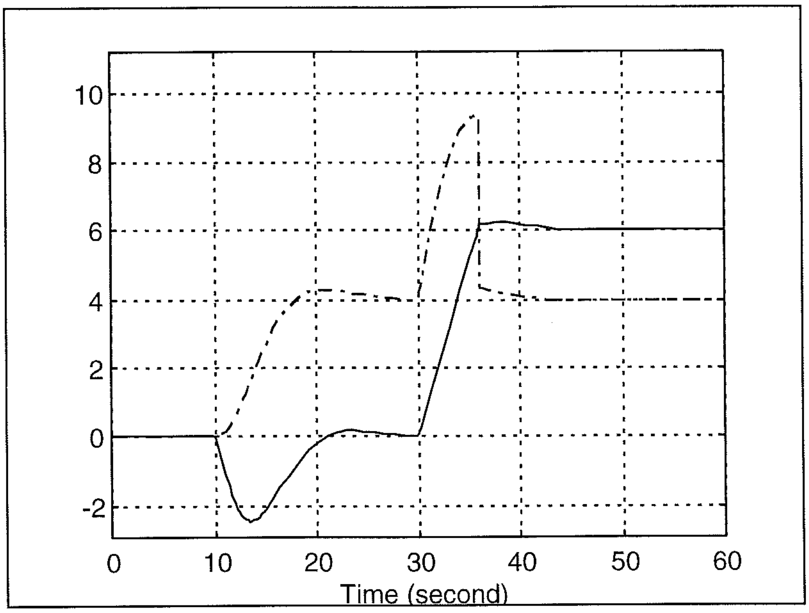


Figure 12
Improved performance of the modified integral discharge on the same kind of disturbance

By moving the set point of the switches it was also possible to make the discharger less susceptible to small changes that can be caused for example by noise or modeling errors.

2.5 Rate Limitations

We will now introduce a very severe rate limitation to the actuator input. Limiting the derivative to +/- 0.2 it takes to the system 10 seconds to fully reverse the control action and 5 seconds to bring to zero the maximum control action allowed by the saturation.

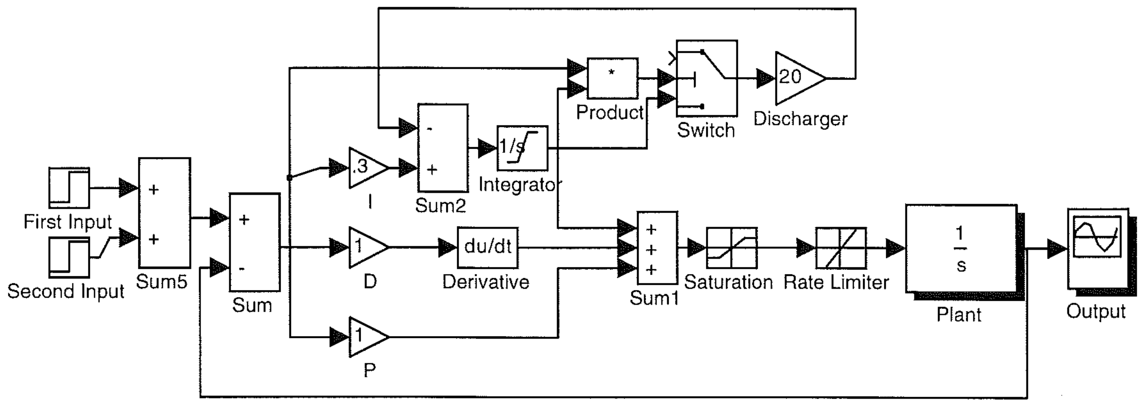


Figure 13
PID with integral discharge affected by a rate limiter

As expected the introduction of the rate limitation causes an evident overshooting since the controller ignores that its actions are delayed in time by at least 3-6 seconds.

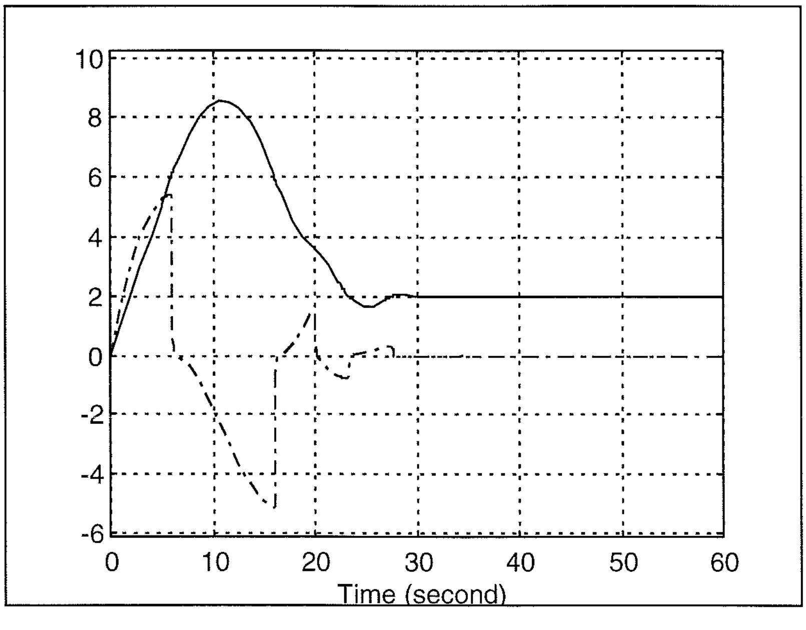


Figure 14
The result of a rate limiter are large overshoots due to the finite time needed to change the input signal

Trying to raise up the value of the derivative action helps a bit when the rate limit is not this much severe but this is not the case. It is evident that some other action is needed to make the controller aware that the input's changing rate is strongly limited.

2.6 Splitting the anti-windup action

When rate limits are not too much severe it can prove a useful solution to split the anti-windup circuit into two parts, giving different weights to the anti-windup activated by the saturation and by the rate limiter.

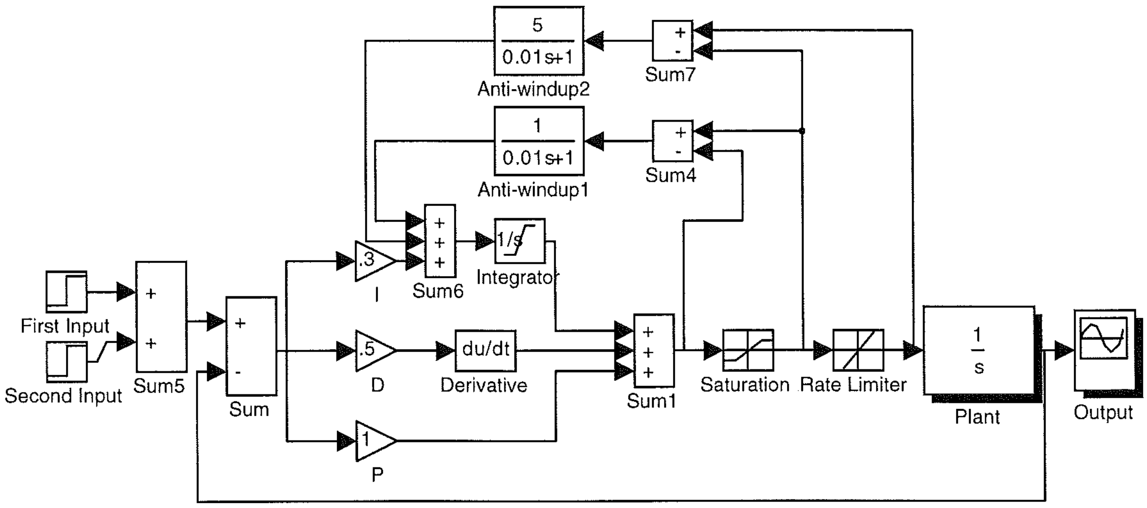


Figure 15

Splitting the anti-windup circuit into two dedicated circuits for the saturation and the rate limiter adds degrees of freedom in terms of parameters that can be set and sometimes allow to tune correctly systems affected by significant rate limiters.

In this way there are more parameters to set and this makes it even more difficult to correctly tune the controller, but the effort is often paid back with much more smooth responses and less overshoots.

Usually good results are achieved by raising the weight of the anti-windup related to the rate limit, however it was difficult to get general rules of thumb.

2.7 Predictor-Feedback

The idea is to manipulate the feedback of the system so that the value is increased by a value that is proportional to the speed at which the system is moving. This means that if the system's state has a certain positive value and the derivative of the status variable is positive, the system's controller will think that the actual system's state is the real one plus an off-set value that is greater or smaller according to the speed itself.

This is simply accomplished by adding a derivative action to the system's feedback. In some case it is useful to add a dead-zone that prevent any anticipating action as soon as a certain speed is reached, since usually the gain of this derivative action as quite high an it can be harmful to have it acting always, changing completely the controller's behavior.

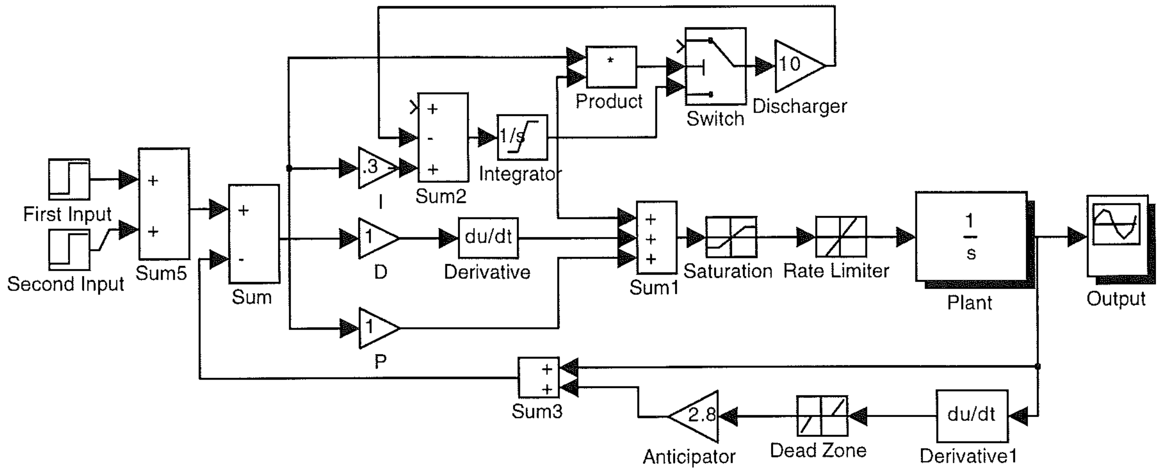


Figure 16
A predictor feedback is added to a standard control loop

Of course, instead of the usual three, we have now five parameters to tune, and it takes some trials to get a desirable response without overshoots.

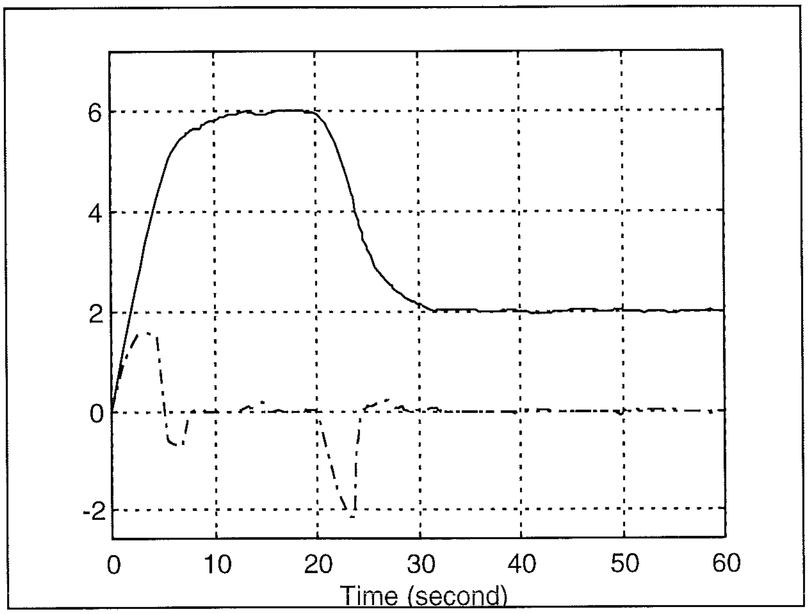


Figure 17
The predictor feedback allow to finely and easily tune PID systems affected by severe rate limits avoiding undesired overshoots

2.8 A Second-order system

In this second example we consider a second order plant in the form :

$$\frac{1}{s^2 + s}$$

that requires some adjustments in the parameters of the anticipator to get a satisfactory evolution due to the more complex dynamic of the plant.

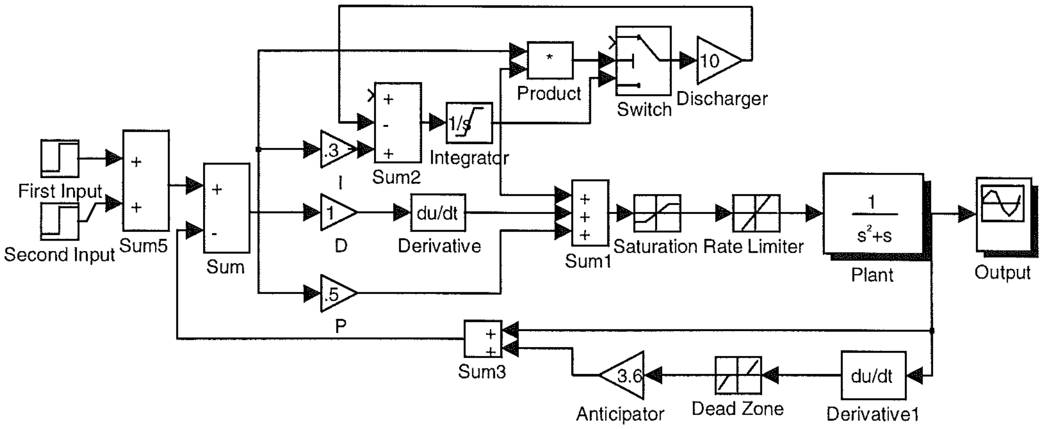


Figure 18
A second order system with a predictor feedback added

Note that, as expected, the integral action is quitted before 5 seconds from the beginning of the simulation even if the system will reach the set-point five seconds later. Raising the anticipator constant too much causes the system to stop too early and in case of a very bad setting even to reverse the direction of the evolution moving far away for a while from the set-point.

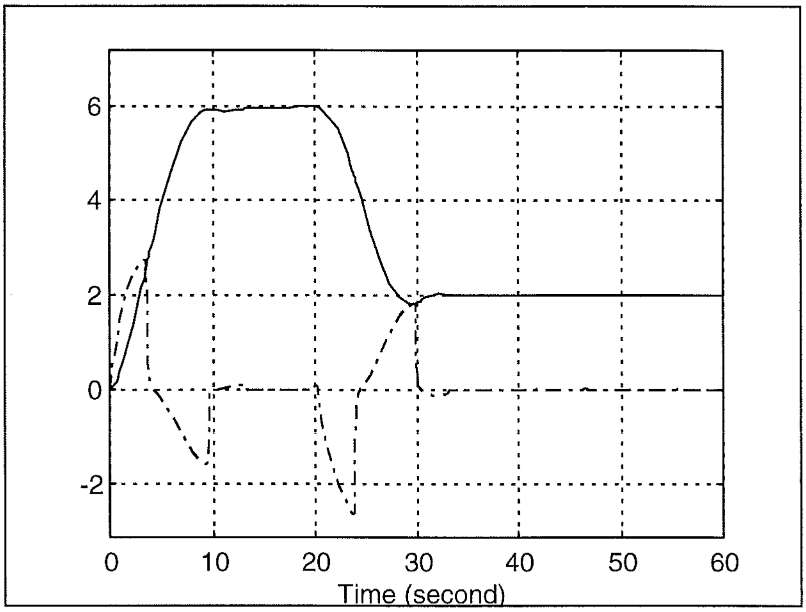


Figure 19
Also second order systems can be easily tuned

2.9 Third order system

A third order system is now considered with a plant's transfer function equal to :

$$\frac{1}{s^3 + s^2 + s}$$

Increasing the dynamic of the system makes it harder to tune it correctly, and this is in general valid for linear systems as well.

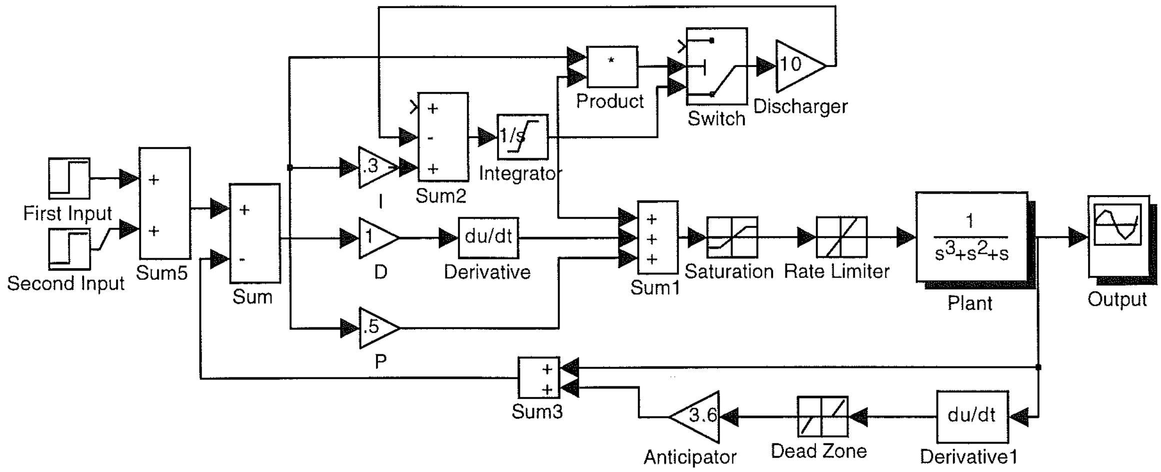


Figure 20
A third order system with predictor feedback

Despite extensive tuning attempts the system reaches the set-point either with sensible delay and/or multiple overshooting ; even if tuned correctly, it is enough to insignificantly change even one of the parameters to fall into a very bad response again.

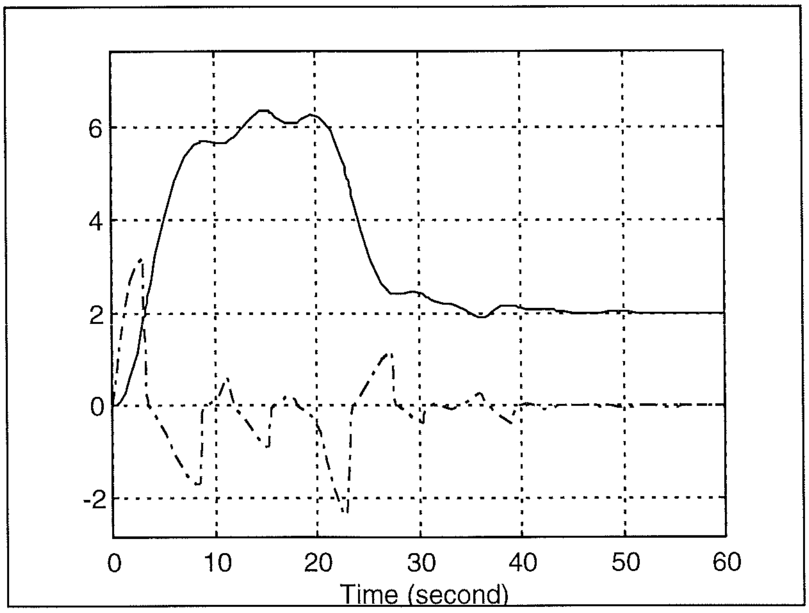


Figure 21
The time response to a third order system is very susceptible to small changes in the parameters

Using an anti-windup system, does not improve much the response. Again it is possible to correctly tune up the controller, but a very small change in the parameters change completely the system's response.

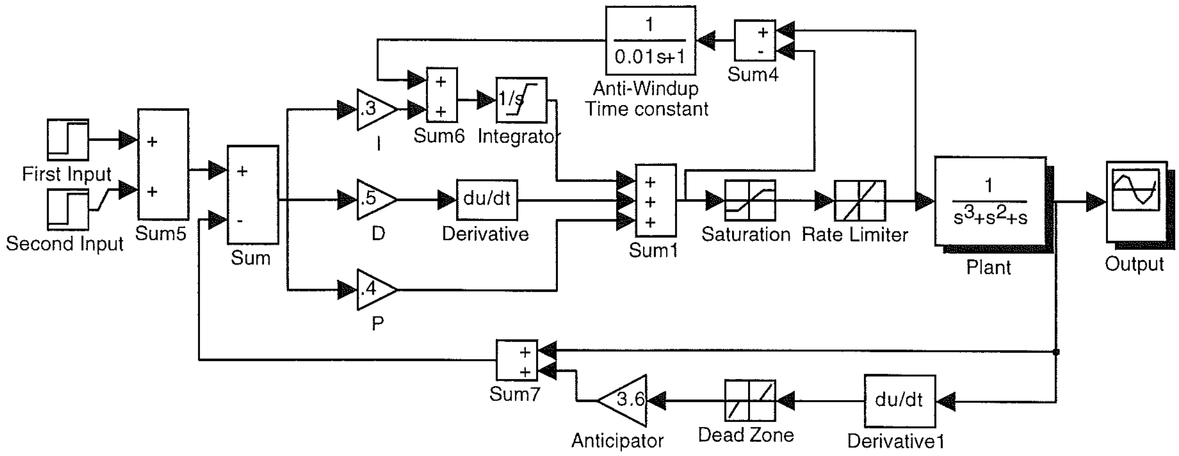


Figure 22
PID controller with anti-windup and predictor feedback controlling a third order plant

The following pictures show the system response to the same input under two different values of the anticipator constant. Even a small change bring the system from a situation in which the problem is an excessive overshoot to another situation in which the system stops completely much before reaching the set-point.

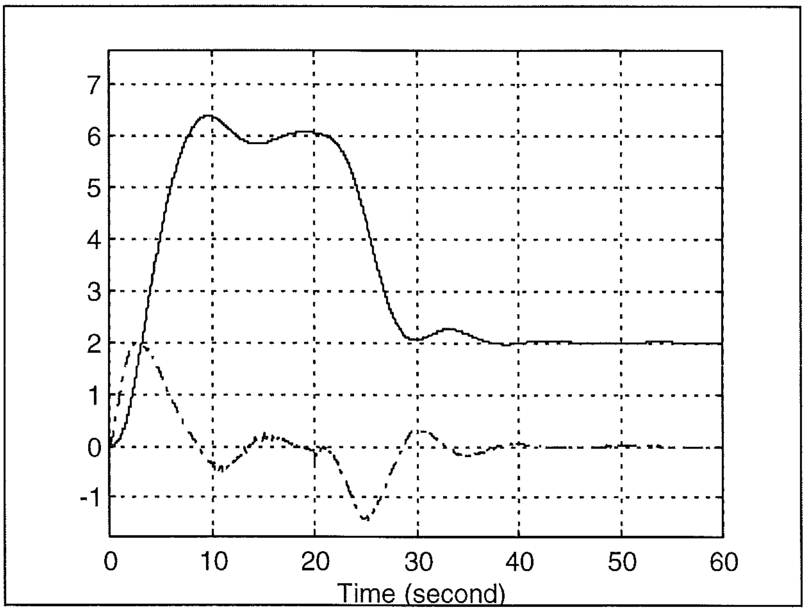


Figure 23
Again the tuning is very difficult and susceptible to minimal changes in the parameters

2.10 Third order system with anti-windup, predictor feedback and integral discharger.

An idea can be to put together all the devices that we examined since now, in a hope that their cooperation will affect different aspects of the problem, making the system less sensible to tuning inaccuracies.

The result is encouraging, since a good system's response can be very easily found but, as stated before, the integrator discharger does not work properly under the effect of external disturbances.

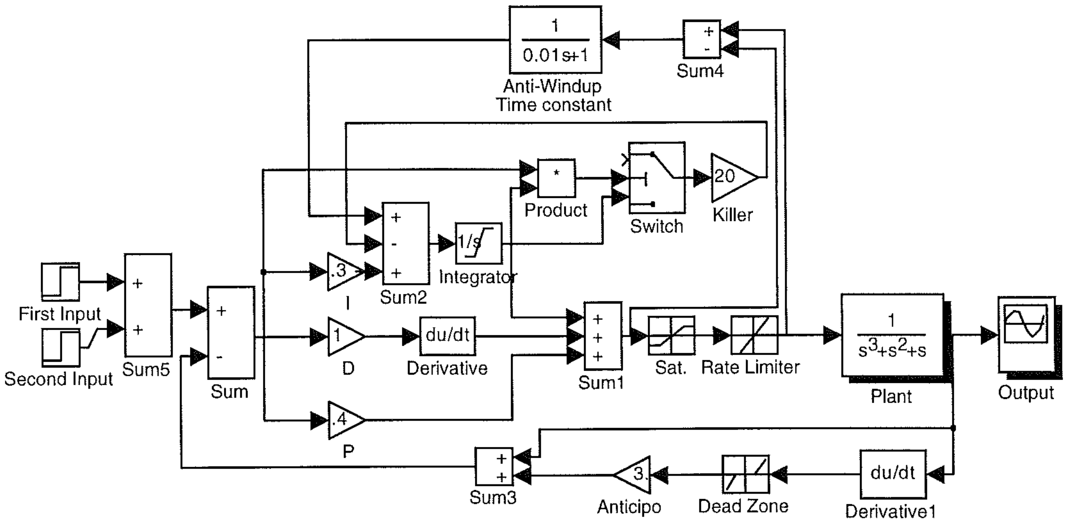


Figure 24
PID controller with predictor feedback, anti-windup and integral discharge

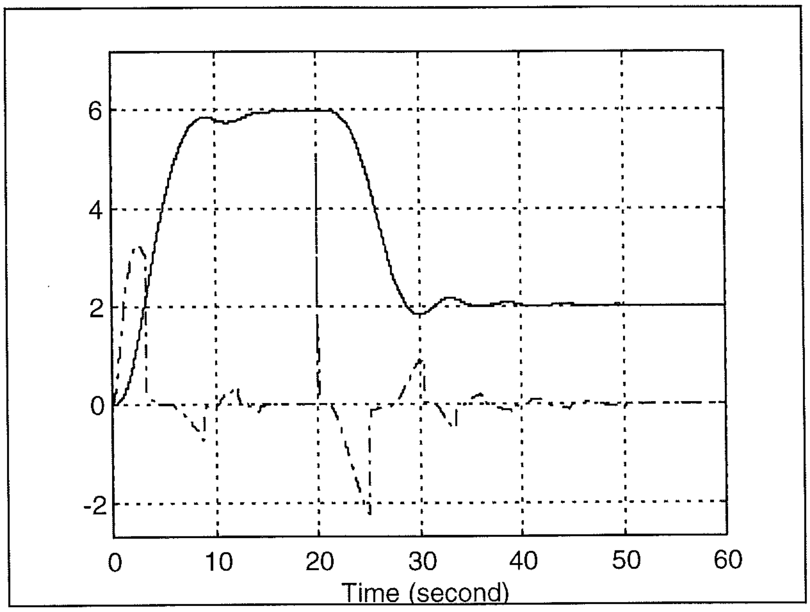


Figure 25
The combined effect of anti-windup and integral discharge makes easy to tune the controller

2.11 Importance of the integral anticipator

The anticipator can also be moved from of the feedback to the controller, acting not only on the feedback signal but on the overall error signal.
Doing so we can split the anticipator into three different parts, to be able to control individually the value of the anticipator constant for each one of the three parts of the PID controller.
Trying now to set a zero gain to some parts of the anticipator leads usually to a deteriorated performance of the system. After some tests it appears evident that the most essential anticipator is the one that acts on the integral action, since it acts on the whole non linear anti-windup system.
In some cases it is also possible to tune correctly the system with only the integral action affected by the anticipator as shown in the next picture.
With more difficult system, or with more severe rate limits it becomes very hard (impossible ?) to find a good tuning with only the integral action affected by the anticipator, giving still the impression that in these case is better to make the whole PID controller affected by the anticipator.
Since the anticipator's most important effect is the one on the integral action, it is not possible to obtain an equivalent system, obtaining an analytical expression and changing the PID parameters according to this linear equivalence, for the reason explained before that the total effect is not linear.

2.12 Conclusions

Anti-windup provide a very good answer to saturation problems. When rate limits are introduced small changes in the tuning or in the anti-windup circuit may provide good compensation of the phase loss. In particular, splitting the anti-windup feedback into two differently weighted parts can be a first step that still does not change the controller too much.
In case of very tight rate limits the extra derivative effect was very effective. The reason is the predictive ability of the derivative. Keeping this concept in mind is also quite easy to tune the anticipator by calculating how much it will make the controller look ahead compared to the typical rising rate induced by the controller and the saturation combined.

3. Phase compensation of rate limiters

3.1 Introduction

During this chapter we will explore some typical effects of rate limiters on marginally stable systems. Under some circumstances a rate limiter in the control loop produces phase lags that can treat the stability of the system ; several filters can be used in the attempt of reducing this phase lag and some of these filters will be compared analyzing the overall problem.
At the end examples will show theoretical improvement of stability and simulated time-response examples to test the various filters.

3.2 Rate limiters and system stability

To introduce the problem focused by this chapter, we will first show an example of a rate limiter reducing stability margins of a linear system.
What follows is a fifth order model, that simulates a fourth order plant and an external operator, represented by a PI controller that controls the system with a feedback loop.

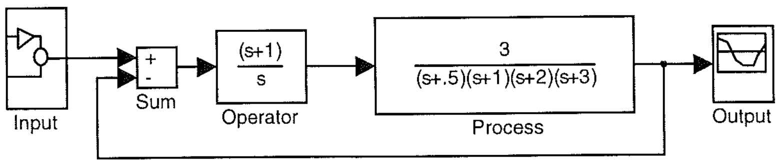


Figure 1
A linear system with a fourth order plant and an external operator

The plant is stable (four real poles with a negative sign), but the combination of the controller and the feedback loop, makes the whole linear system marginally stable ; stability margins are very little and therefore a little increase in gain can make the system unstable.
There are many examples in the physical world of unlucky couplings of a stable plant with an external (typically human) operator which result into an unstable or marginally stable system.
Anyway the considered system is still stable ; now the response of the system is illustrated, the input signal chosen is a ramp that starts at zero value and ends up at a value of three and remains constant thereafter.

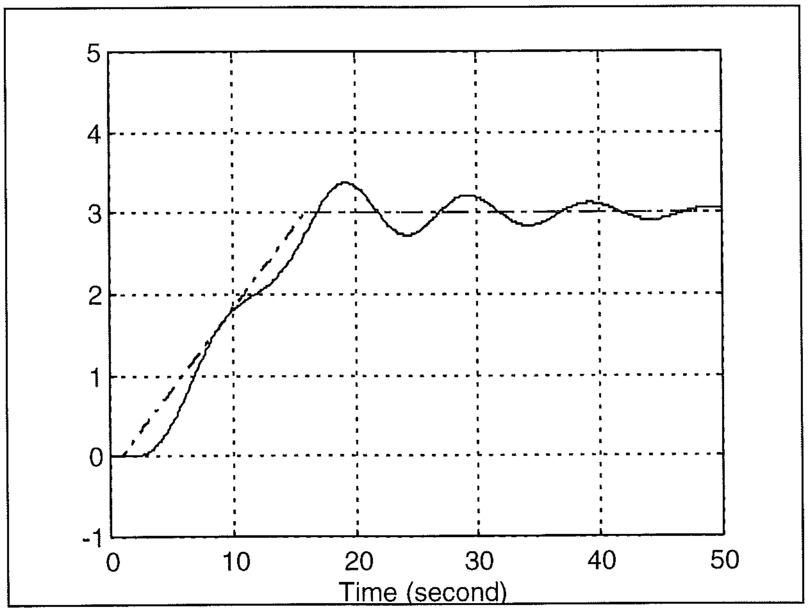


Figure 2
Time response of the linear system (solid line) to a gradually increasing input (dash-dotted line)

As expected the output follows the input with large oscillations due to small stability margins ; anyway on the steady state the output will reach the input.
Now we will introduce a rate limiter between the operator and the plant. The overall system is no more linear and therefore it is not immediate to predict the behavior of the system.

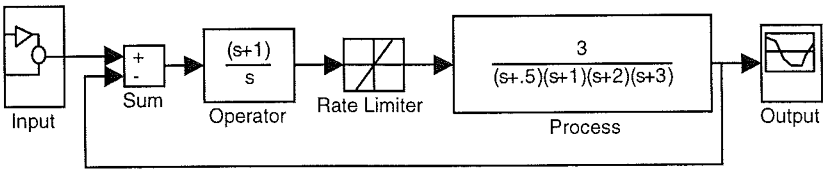


Figure 3
The modified system with a rate limiter in the control loop

The time response of the new system to the same input does not show any significative difference ; the reason is that the slow input results in a small rate of change in the signals inside the loop. The rate limiter is sometimes exceeded, but only for very brief moments and the overall time response, showed in the next figure, is almost identical to the previous one.

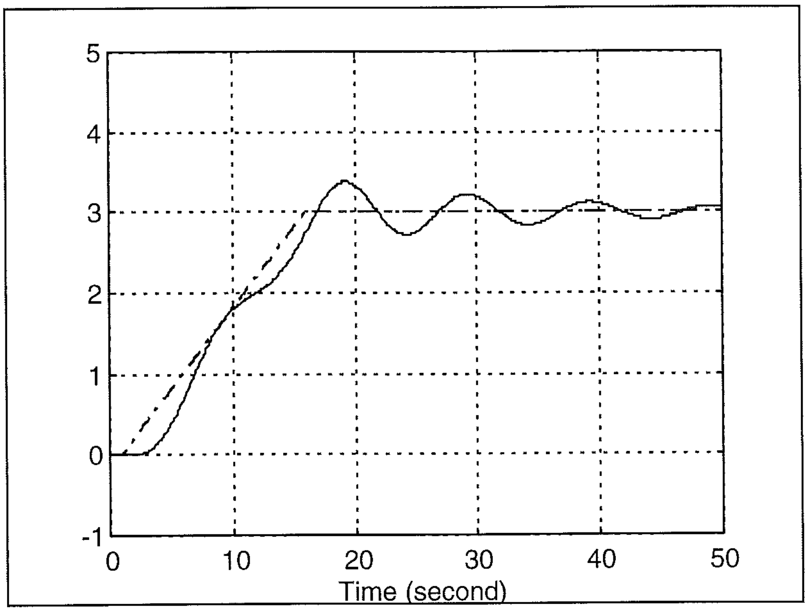


Figure 4
Time response of the system with a rate limiter (solid line) to a gradually increasing input (dash-dotted line)

If, instead of a gradual input, we choose a faster input that reaches again a value of three after only one second, instead of the 18 seconds needed by the first input considered, we obtain drastically different responses.

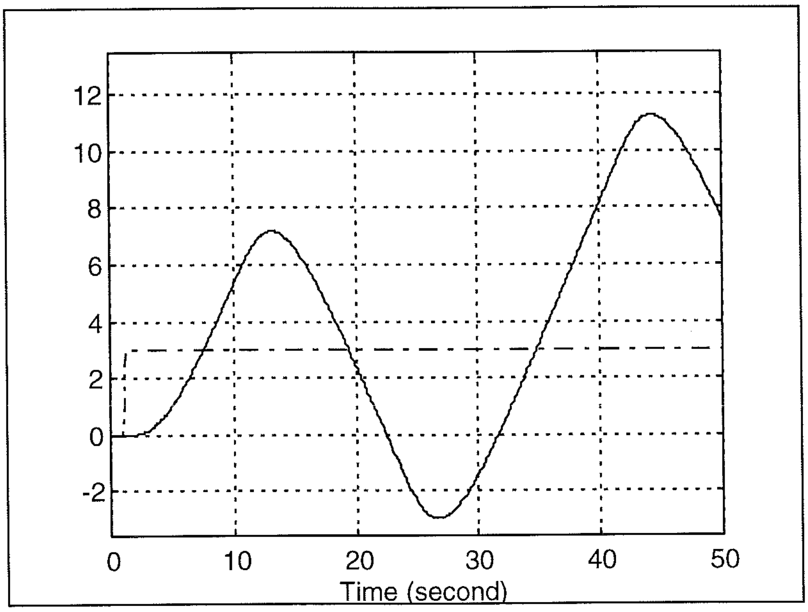


Figure 5
Time response of the nonlinear input (solid line) to a rapidly increasing input (dash-dotted line).

The response of the linear system is always stable, and this is proved to be true for any input regardless of magnitude, bandwidth or any other parameter that can be chosen ; on the other hand, is very interesting to observe the time response of the non linear system, shown in figure 5.

It is clearly visible that the system is diverging around the set point and the control loop fails to stabilize the plant.
On the steady state, since the system is not linear, the oscillations will not increase forever but will reach a given level, without converging to the desired value.
In the next paragraphs reasons for this phenomena will be illustrated.

3.3 Representations of a Rate Limiter

There are several ways to represent a rate limiter. As already mentioned, software rate limiters are used to prevent actuators from reaching their physical limits and therefore their effect is dominant on the dynamic of the system.
Given a sampling time T and a rate limit R, the simplest form of software rate limiter keeps a buffer with the last output value Y1 and compares it with the new input U received.
The corresponding output will be equal to the input only if the condition :

$$\left|\frac{Y1-U1}{T}\right| \leq R$$

is true, otherwise the new value is limited so that the previous condition is verified.
In a continuos system, a possible representation of a basic rate limiter is given by the following block diagram.

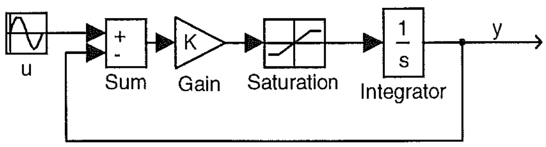


Figure 6
A block diagram that behaves like a rate limiter

the smaller the gain 'k' closer to a theoretical rate limiter will be the behavior of this system.
Usually it is not convenient to have a too high value of k due to stability problems, since it will be impossible for the limiter to keep a constant value without starting to swing around a mean value at the maximum rate and at a high frequency.

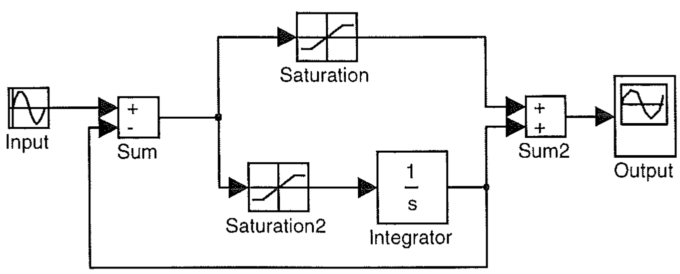


Figure 7
Another common form of an analog rate limiter

In practical implementations is also commonly used the block shown in figure 7 where small signals are fed through and larger ones are rate-limited. This is also another way to make the rate limiter be influenced only by really significant signals that exceed over a certain value while small ones are unaffected.
Notice that in this case also the overall behavior is somewhat different than the one of a theoretical rate-limiter.

3.4 Non-linear analysis

The behavior of a rate limiter is obviously non-linear. Therefore we can not restrict our analysis to simple responses like an unitary step but we will have to test instead a greater range of inputs.
If we feed a rate limiter with a sinusoidal wave of a given amplitude A and frequency f:

$$u = A\ sin(2\pi\ f\ t + \varphi)$$

we can expect four major kind of outputs that all depend on the frequency and amplitude compared to the rate limit R of the rate limiter.
If we derive the signal input we obtain the expression :

$$\frac{du}{dt} = 2\pi\ A\ f\ sin(2\pi\ f\ t + \varphi)$$

that gives us a clear relation between A, f and R. In fact if the condition :

$$2\pi\, A f \leq R$$

is verified the output Y of the rate limiter will be the same of the input at all times ; in this case the rate limiter will have zero influence on the signal.
If instead this limit is slightly exceeded there will be some values for which the output signal will not be able to follow the input signal and temporarily the two values will be different. This is the case showed in the next figure. Notice that the output signal meets again the input signal before the extreme is reached and therefore there is not yet any gain loss.
If we calculate the first armonic of the output signal we can already notice a small phase delay, due to the asymmetry of the output curve, but still the output reverses direction at the same time with the input.

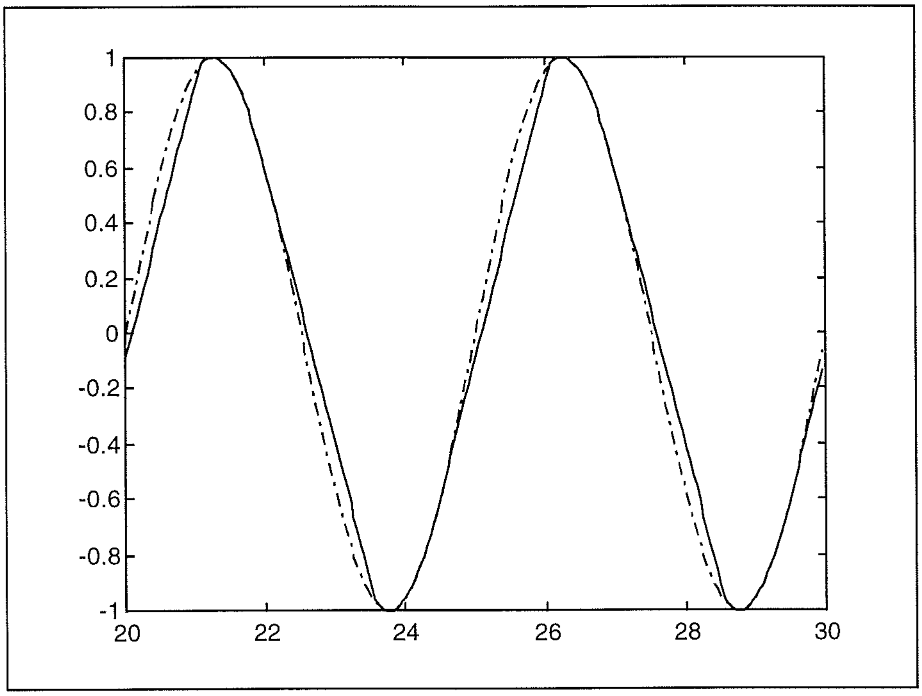


Figure 8
A rate limiter starts to change the shape of a signal

If we increase a bit more the frequency or the amplitude of the input we will soon fall into a third case in which the output signal is no more able to reach the input signal before the extreme of the curve. At this point we start losing gain and the amplitude of the output will be less than the amplitude of the input.

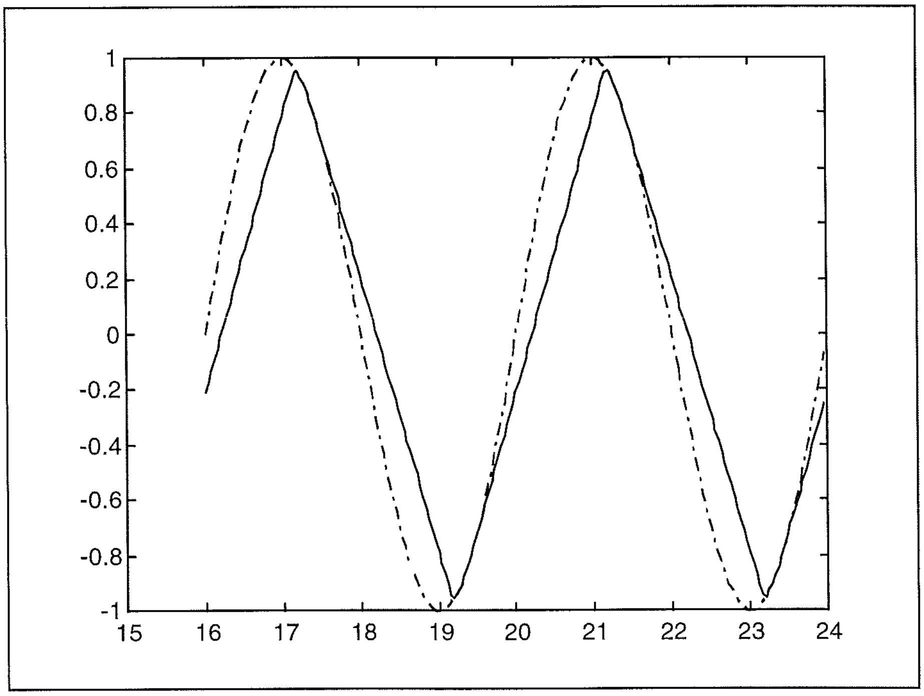


Figure 9
An increase in frequency and/or amplitude of the input signal leads to a visible loss of phase and gain.

The shape of the output is getting progressively closer and closer to a simple triangle wave and only a small part of it is still following the sinusoidal wave. This can be clearly seen in the previous picture where, as usual, a unitary sinus wave is fed as input and the frequency is carefully tuned to obtain the desired response.
If we increase slightly the frequency or the amplitude of the sinus wave we fall into the last possible kind of response ; in this case the output signal will always try to chase the desired output without reaching it but during crossovers.
The response will be a perfect triangular wave, and increasing the frequency of the amplitude of the input will lead to an increasing loss of phase and gain. The phase lag will progressively increase to a limit of 90 degrees that will of course never be reached.
The following figure shows this extreme fourth case that will be mostly under attention during the rest of this chapter.

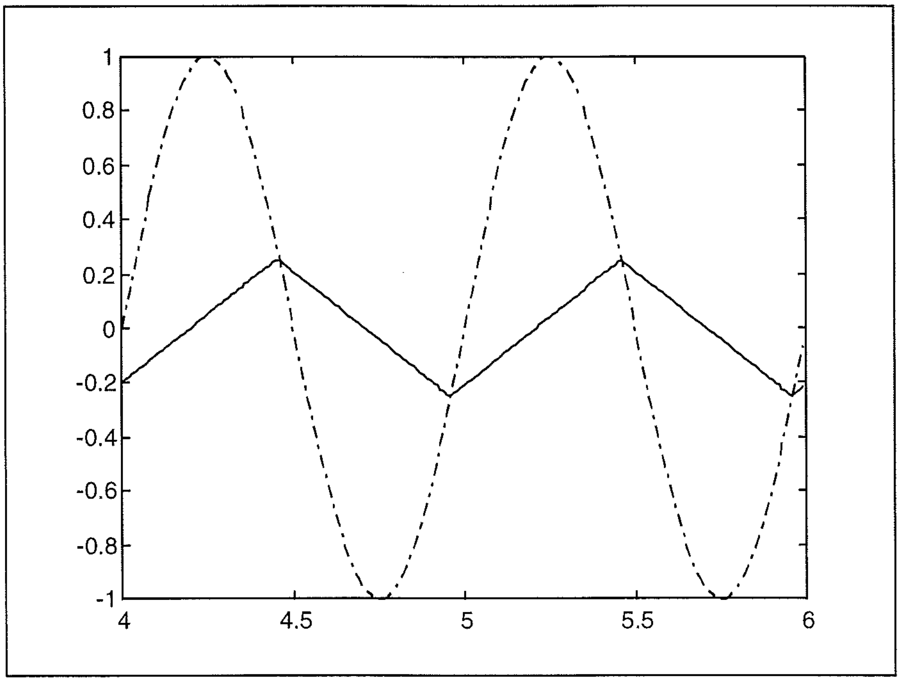


Figure 10
High frequency of high amplitude response of a rate limiter to a sinusoidal input is a regular triangular wave.

It is important to notice that the amplitude of the output signal depends only on the frequency of the input wave according to the expression :

$$A_{output} = \frac{R}{4f}$$

therefore the amplitude of the input wave has no influence on the output.
If we consider the gain to be a simple ratio between output and input, we can obtain a very easy expression that describes the gain as a function of the input amplitude and frequency.
Given :

$$Gain = \frac{A_{output}}{A_{input}}$$

after the substitution we can obtain the simple formula :

$$Gain = \frac{R}{4f\ A_{input}}$$

that again shown that frequency and amplitude of the input contribute with the same weight to the deterioration of the signal's gain.
Also the phase lag can be analytically calculated with a less elementary procedure but before proceeding further the next figure gives an idea of the graphical criteria that were used to calculate the previous formulae :

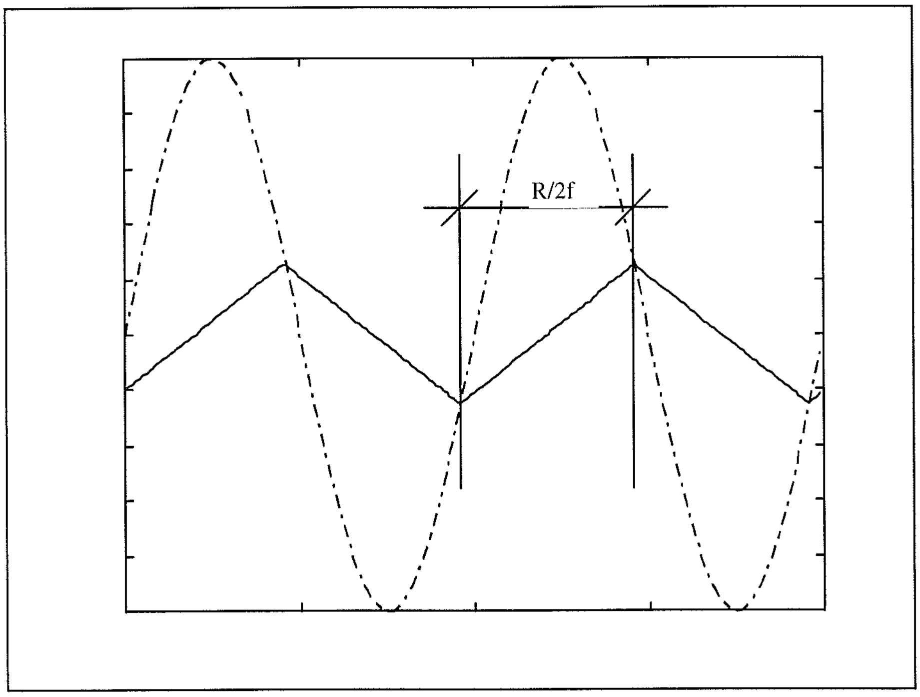


Figure 11
Graphical calculation of gain

of course the formula that describes the gain makes sense only of the resulting gain is less than one.
To compute the phase-lag resulting from the rate limiter in this fourth case we start with a similar graphic approach.
Translating the output wave so that its maximum takes place when the sinusoidal wave has a zero value, we can see that the amplitude of the output signal (known) is equal to 90 degrees minus the sinus of the angle of phase lag (unknown).
In formulas :

$$phase = arcsin\left(\frac{R}{4f\ A}\right)$$

in the same way as stated for the other formula for the gain, also this formula makes sense only when the other one does. In this case the limited validity of the arc-sinus function helps to prevent misuse of it.

In the following diagrams gain and phase-lag are plotted against a logarithmic frequency scale using the previous formulas. The system considered is characterized by a unitary rate limiter and the various curves show different responses to input waves of 10 different amplitudes, from 0.5 to 5, as shown by the arrows on the graphs. As expected the gain tends to zero, while the phase-lag tends to 90 degrees.

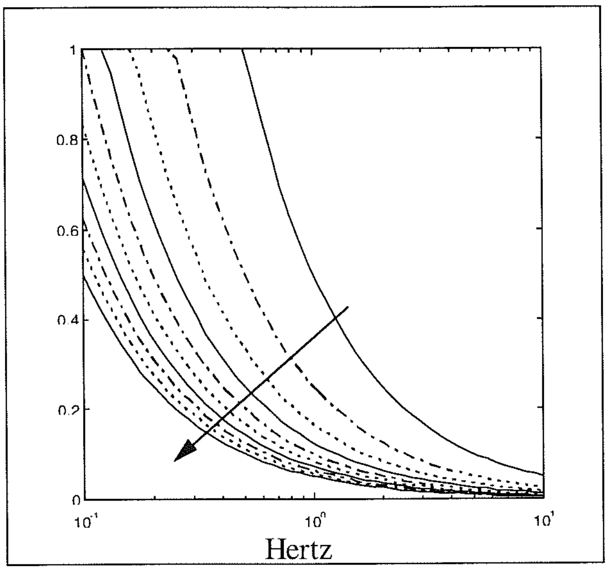


Figure 12
Frequency response (gain) of a rate limiter for different values of the input amplitude (range from 0.5 to 10)

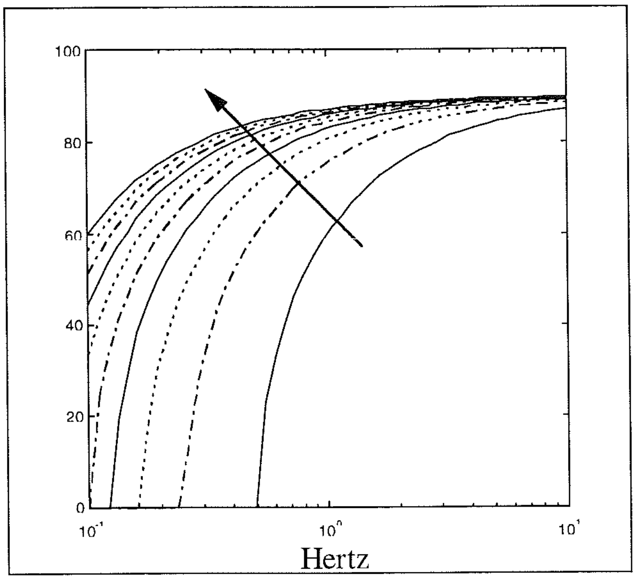


Figure 13
Frequency response (phase loss) of a rate limiter for different values of the input amplitude (range from 0.5 to 10)

Another way to obtain similar diagrams is through simulation. The following two diagrams show gain and phase-lag obtained through multiple simulations at different frequencies. For each frequency a simulation was performed for 200 periods, to ensure that steady state was reasonably reached, and then the last periods were analyzed to obtain the gain and the phase lag compared to the input wave.
The resulting graphs are very similar to the ones obtained with the analytical formulas except for the small transitions between the unitary gain and the hyperbolic curves due to the fact that only the fourth case is calculated by the formulas.

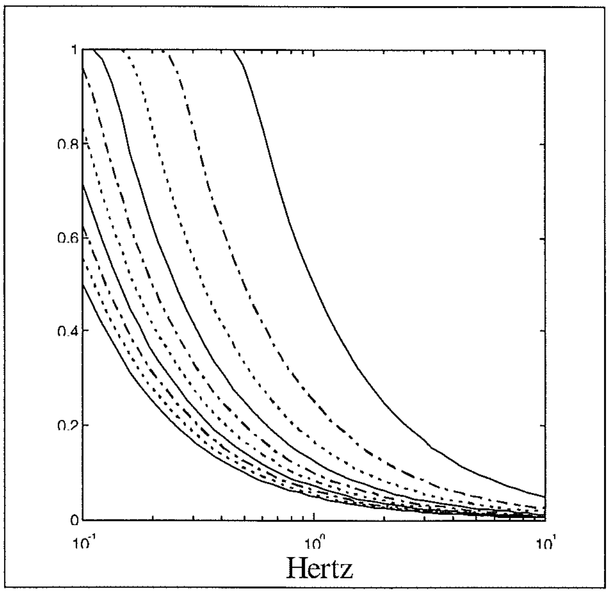


Figure 14
Frequency response (gain) of a rate limiter calculated through simulation

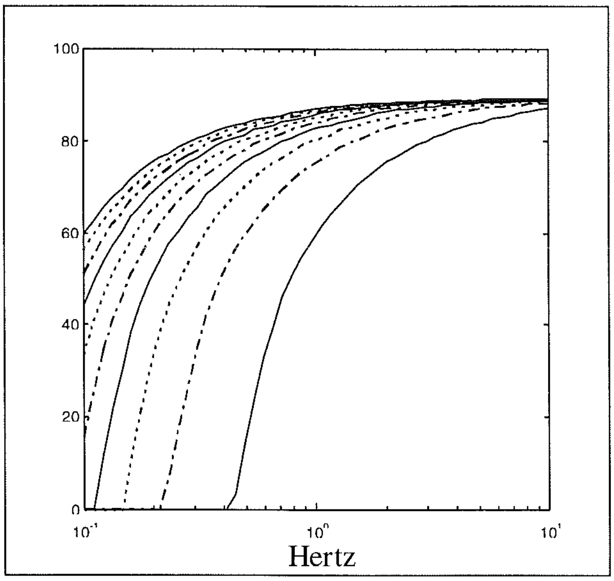


Figure 15
Frequency response (phase loss) of an unitary rate limiter calculated through simulation

As a result of this, the 'real' curves show a smooth transition that is not described by the analytical ones.
To obtain a sharp analytical solution is necessary to calculate for all frequencies the descriptive function of this non-linearity, operation already done in literature and that brings to the same curves that were experimentally found through simulation.
In the next sections several non-linear systems of increasing complexity will be considered and therefore it will be in many case impossible (or too complicated) to calculate a descriptive function for them and simulation will be the only tool available to obtain a good and reliable frequency response.
Independently from the system used to analyze the behavior of the rate limiter it is evident that for high frequencies the phase-lag becomes more and more important, up to a maximum value of 90 degrees.
If the rate limiter is part of a control loop, this means that for some high frequencies or high input amplitudes the phase margin (or gain margin) of the control loop will dramatically decrease and in some case will lead to instability.
For aerodynamically unstable planes the Flight Control System does not provide more than 45 degrees of phase margin and it is quite clear that this limit may be easily exceeded.
Typical PIO (Pilot Induced Oscillation) accidents occurred in concomitance with strong and sudden manual inputs that saturated the rate limiters. This decreased the phase margin changing the behavior and response of the system, leading to even more diverging inputs.
In the following section different phase compensation strategies will be discussed to prevent or reduce this phenomena.

3.5 Phase compensating filters

The presence of a rate limiter changes in some cases the signal and we showed in the previous paragraphs that this change results into a loss of gain and phase.
It is quite obvious that there is nothing to do with the gain loss ; simply the input's rate of change is too fast for the limiter and the corresponding evolution of the system is limited.
On the other hand, looking at the previous figures it is possible to think about phase compensation and we can expect the possibility to design filters that reduce the phase lag when the rate of change of the signal is high enough to saturate the limiter.
The phase lag is also the most dangerous phenomena associated with rate limiters, since it endangers stability in a closed loop, while the gain loss is not usually associated with stability problems.

3.6 Phase compensation using feedback

This method is described in [7], [8], [9] and [10] and gets it is main idea from anti-windup methods. An anti-windup logic is a circuit, used in a PID controller that feeds back a negative signal when a saturation limit is reached and uses this signal to discharge the integral whose state variable is building up an excessive value compared to the saturation itself.
Analyzing a frequency response of an anti-windup system, it is easy to notice a phase advance when the non linearity is taking action.
The idea is to use a feedback around the rate limiter that feeds back the signal to a low pass filter and from here to the signal source ; in presence of a rapidly changing signal, the anti-windup circuit will start to build up a negative term that affect increasingly the overall signal.
When the input signal reverses the direction, the negative component causes a premature reverse of the output signal and thus reduces the phase lag.

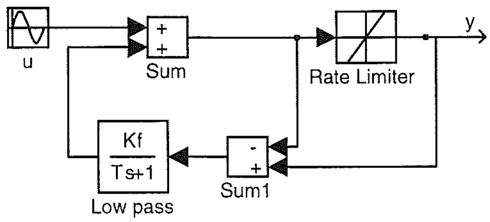


Figure 16
Rate limiter with feedback

Theoretically the best choice would have been to have an integrator instead of the low-pass filter, but this does not prove to be a stable choice in the real use and therefore the low pass filter is preferred.
Analyzing this non-linearity's response to different frequencies and gains we obtain a frequency response diagrams like already done with the basic rate limiter.
These two diagrams are shown in the next two figures (gain and phase) for various amplitudes of the input signal (0.5, 1, up to 10).
Notice that since the filter is not linear, if we have a frequency response to an unitary gain, we can not automatically know the responses to all the amplitudes, and this is clearly shown by the diagrams that show completely different responses for different amplitudes.
Again, knowing the frequency responses for any frequency and any amplitudes does not imply that we know the response of the filter to any input, still the non-linearity can behave unpredictably to inputs of other kinds. In this never ending quest to crash-test the various non-linear filters other tests will be done, testing the filters on time simulations with various inputs.

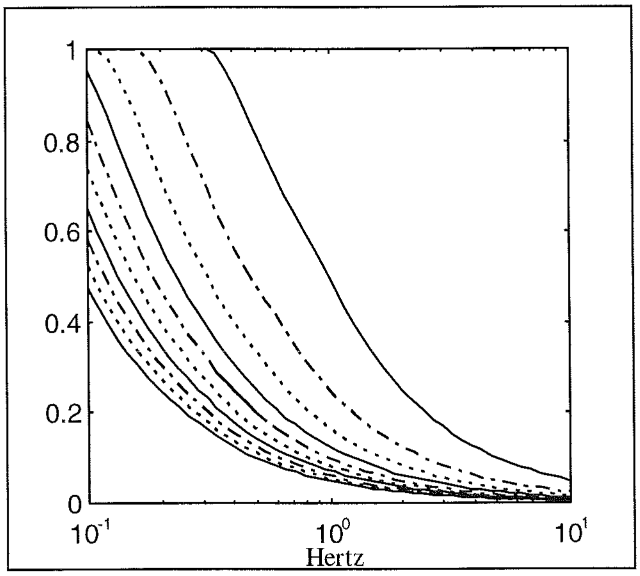


Figure 17
Frequency response (gain) of a rate limiter with feedback

If we compare the gain loss with the one of a standard rate-limiter we see little or no difference. The reason is that, as previously explained, there is little to do with gain loss.

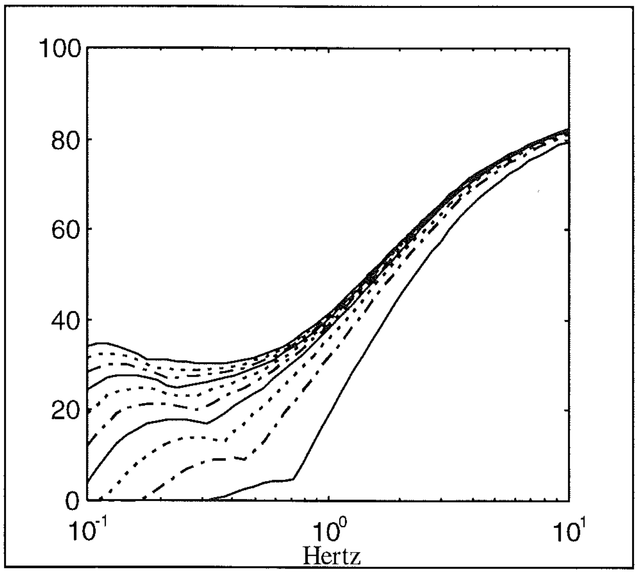


Figure 18
Frequency response (phase loss in degrees) of a rate limiter with feedback

On the other hand, the phase loss is dramatically different. This time the phase loss remains small for almost another order of magnitude of the input frequency, after which, again, tends to 90 degrees asymptotically.
To a frequency analysis this filters improves the phase loss still providing an input limited in terms of rate ; however during deeper tests it was found by the authors of the filter that high frequency signals tend to jam the filter.
From the frequency analysis it is clearly visible that there is no compensating effect for high frequency signals, therefore high frequency components of the signals tend to activate the filter loop even if this will be totally useless, decreasing the performance to low frequency components of the signal.
To compensate this it is possible to add a by-pass for high frequency signals, using the following modification to the block diagram :

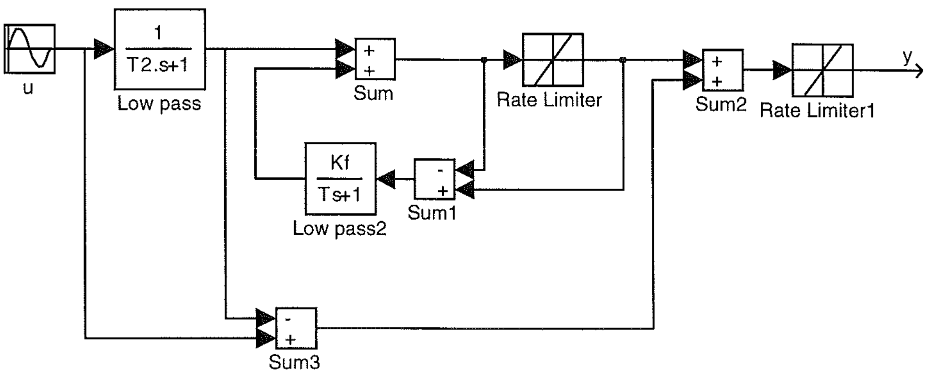


Figure 19
Rate limiter with feedback and by-pass

In this case a low pass filter divides the signal, feeding the compensating filter only with low frequency components of the signal. At the same time high frequency components are bypassed and added just before the output.

Frequency analysis of this final version of the filter is shown in the following figures ; the shapes of the curves this time reserve some surprises.
The gain, as usual is not too significant, gain is progressively lost for high frequencies and high amplitudes of the input signal. It is worth to notice anyway that this time the curve shape is somewhat different and a small plateau is more visible for high input amplitudes.
On the overall the general gain is somewhat slightly smaller than the original one ; this phenomena is common more or less to all phase compensating filters and is due to the fact that filters tends generally and independently to their functioning logic, to anticipate the moment at which they should reverse the signal. If this is not done at once and with the maximum rate instantly, this sometimes results in a smaller evolution.
Again, for stability concerns we will be more focused of phase loss, and this small gain losses, present also in the future filters, will always be very small.

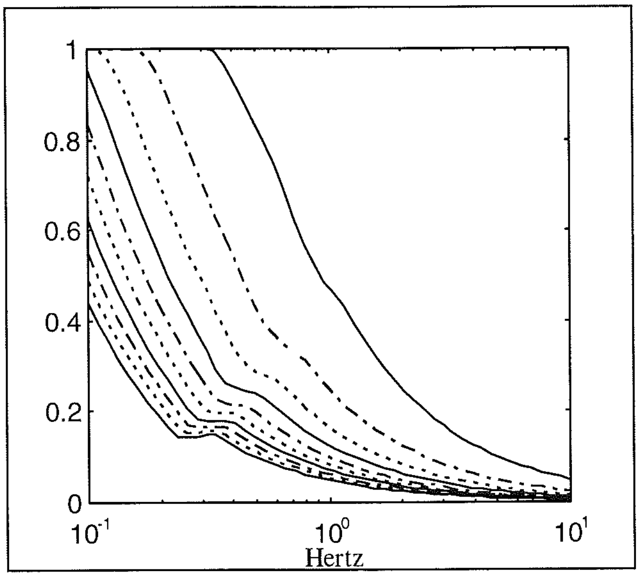


Figure 20
Frequency response (gain) of a rate limiter with feedback and by-pass

The phase loss presents instead a dramatically different shape than the two previous ones which were characterized by regular and smooth behaviors.
As the previous filter, the curves of this one increase significantly the range of frequencies at which the phase lag remains small ; in addition for each input amplitude there is a small range of frequencies in which the phase compensation has a positive peak and the phase lag resulting is very small if not zero ; in this small range of frequencies the performance of the filter with by-pass is of course much better than the one of the previous one.
High input amplitudes results in increasing peaks that end up into negative phase lags ; this, for stability requirements is, of course, a positive and desirable occurrence since the filter will compensate for phase lags of other components of the system ; however this also means that the filter is acting in a non-causal way, literally acting before the input, and this of course leaves some perplexity at a first glance. In the following paragraphs we will see that occasionally negative phase lags are a common occurrence of many phase compensating filters.

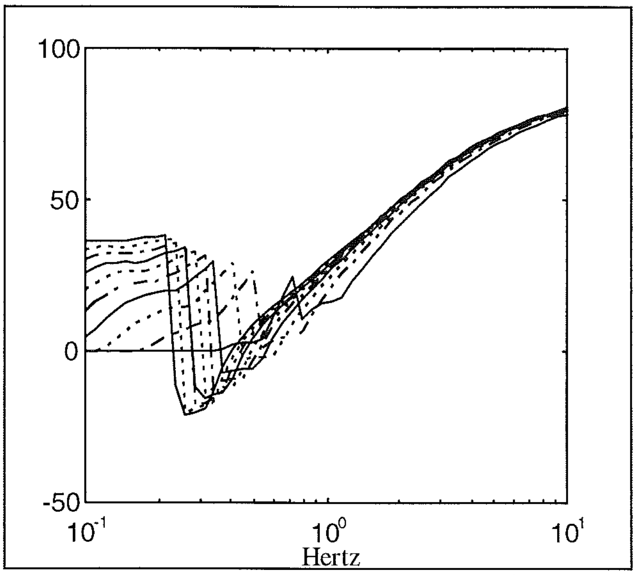


Figure 21
Frequency response (phase loss in degrees) of a rate limiter with feedback and by-pass

The negative phase lags is usually not a major concern due to the fact that together with the rate limit, usually also the overall input action is saturated up to a certain value, therefore on most systems these negative phase lags will never occur.

3.7 Phase compensation using derivative action

Phase compensation can be obtained in several ways. The new filters presented use feedback to compensate phase loss and provide good phase compensation up to a limited frequency range. Other ways can be used to compensate phase loss and in the following paragraphs two of them will be analyzed and compared with the previous ones.
A way to obtain phase compensation is through derivative action ; in conventional PID controllers, the derivative action usually provides reduction of phase lag and increase of overall stability.
The problem is that derivative action itself causes also a change in the shape of the signal and not always this phase advance is desirable ; therefore it is necessary to have a derivative action that dynamically is switched on and off.

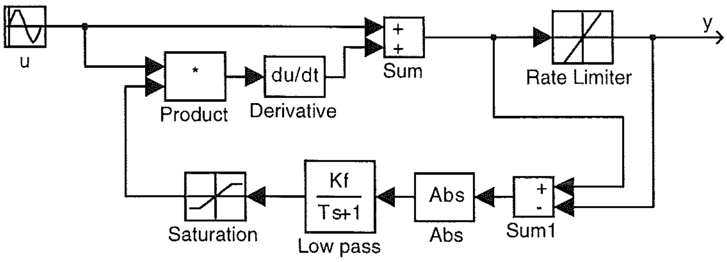


Figure 22
Rate limiter using derivative action

This problem is solved through the use of a feedback signal proportional to how much the rate limiter is exceeded. This feedback signal "charges up" a low pass filter and the

resulting signal is used to influence proportionally the weight of a derivative action that adds itself to the incoming signal stream.
The saturation at the end of the low pass filter is used to impose a maximum value to the derivative action.
In this way whenever a fast signal reverses direction, the tracking signal will reverse soon afterwards due to the presence of the derivative action that will make it "look ahead".
The next figure shows as usual frequency response of the filter to a wide frequency range and for different values of the input signal :

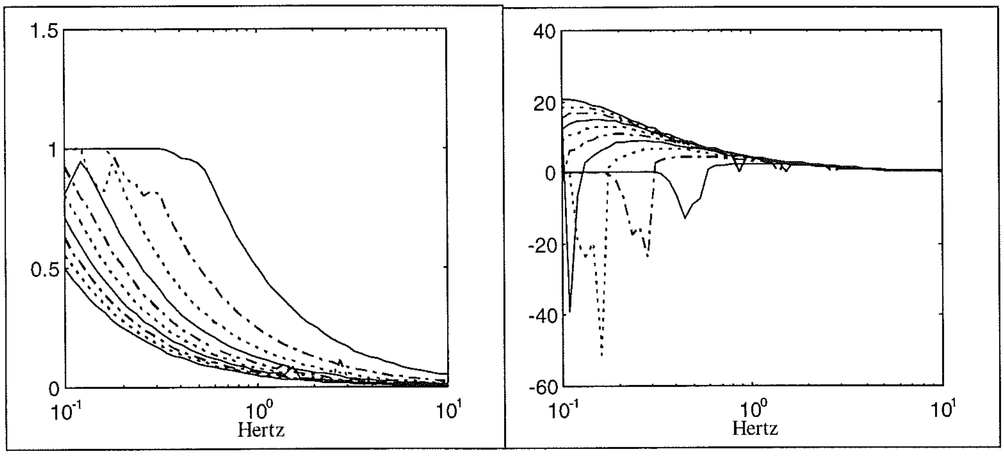


Figure 23
Frequency response (gain and phase loss) of a rate limiter with derivative action

These pictures reserve some surprises that require some explanations.
First of all we notice that the filter behaves better for high frequencies and shows instead a decrease of performance (though very little) between 0.1 and .5 hertz, gap clearly visible with high input amplitudes.
The reason for this is that for small frequencies the derivative of the signal is generally small and may take a long time to reverse the signal, meaning that the derivative action does not make the tracking signal to "look ahead" enough.
This problem on the other hand does not occur for higher frequencies where, for an equal amplitude of the signal, the derivative value is bigger.
The following two figures show the frequency response in a bigger frequency interval, so that it is more clearly visible that the lack of phase compensation occurs in a limited band of frequency that goes back to zero for both higher and lower frequencies.

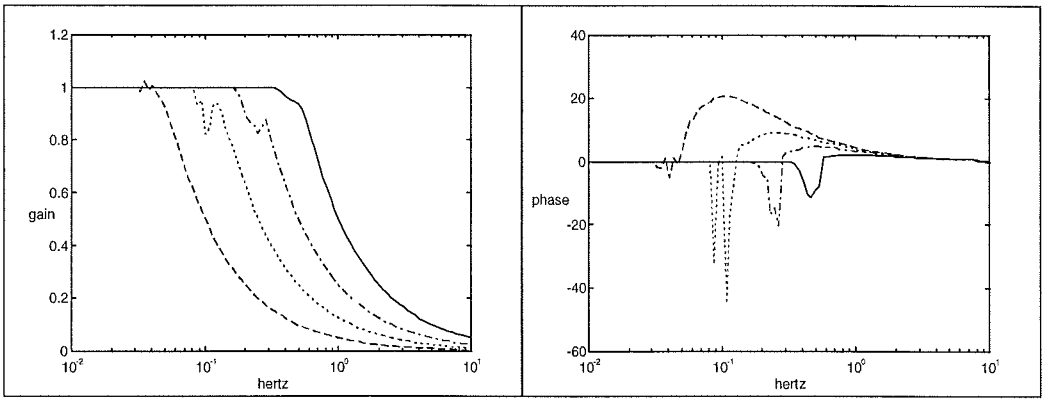


Figure 24
Frequency response (gain and phase loss) of a rate limiter with derivative action

Another thing easy to notice are the occasional irregularities in the shape of the curves, with high and sudden decreases in the phase lags, with seldom negative values. Corresponding to these occurrences, the gain diagram shown also smaller irregular patterns.
To understand the reason for such irregularities it is important to recall again the way these curves are plotted : for each point a simulation is performed up to usually 100 periods, to guarantee that steady state is reached and afterwards the last periods are analyzed to extract information upon gain and phase lag.
Zooming the frequency range with irregularities, shows a different shape of the curves and changing sampling points with closer ones changes again the shape of the curve. The conclusion to all this is that we are facing a sort of numerical instability and the system has more than one possible steady state that can be reached according to different initial conditions.
Observing again the block diagram we can clearly see the "bug" that causes this erratic behavior : a high rated signal switches on the derivative effect, but this derivative then feeds back the switch, and this can cause for example that a signal that is no more violating the rate limiter continues to be influenced by a no more necessary derivative action.
During the early design stages of the filter, this effect was thought to be desirable, but the previous diagrams clearly show that it has to be avoided.
The block diagram can therefore be modified so that the derivative switch is not influenced by the overall signal but instead only by the input signal.

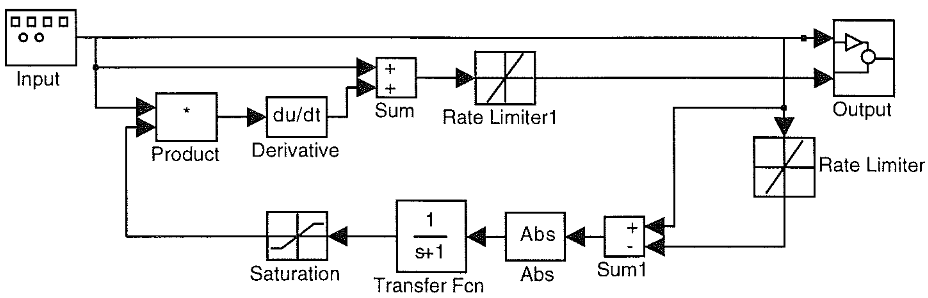


Figure 25
Improved rate limiter using derivative action

The following figures show the behavior of this new rate limiter as usual for different input amplitudes and over a wide range of frequencies.

The overall shape of the curves remain the same but this time there are no more instability effect. There is instead a strange transition phenomena that occurs when the frequency is close to the value at which the derivative action starts to be activated.

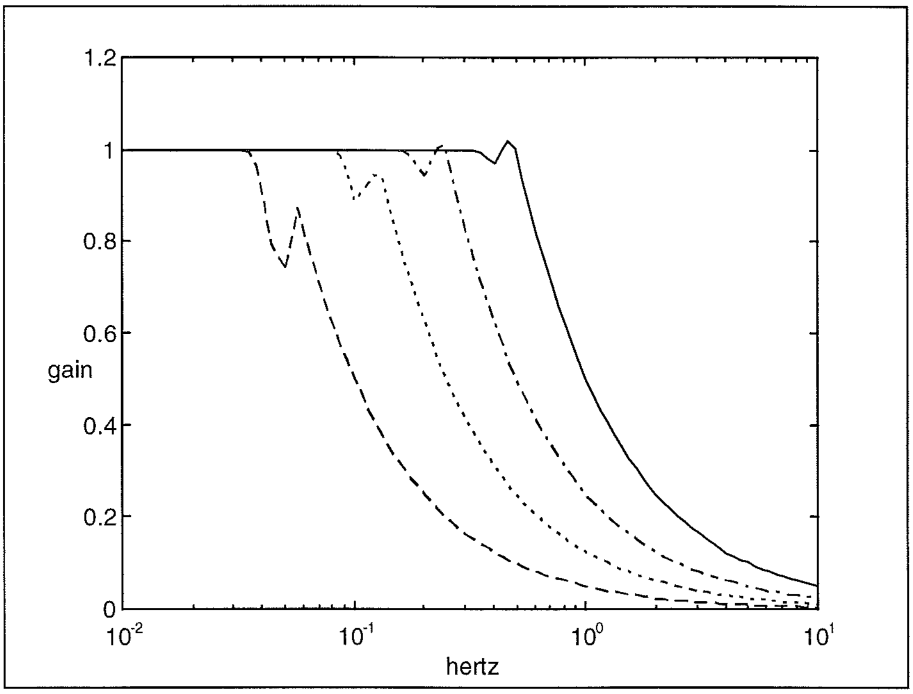


Figure 26
Frequency response (gain) of the modified rate limiter with derivative action

The phase lag diagram shows a sudden loss of phase, followed by a sudden gain of phase. These two abnormalities last for a very little range of frequencies, after which the behavior goes on as expected.

Further improvements can be done to smoothen this transition and avoid this irregular behavior, notice also that there are a lot of parameters that can be changed in this filter, therefore there are several degrees of freedom for it is tuning such as the low pass filter gain and time constant, the value of the maximum values allowed by the feedback saturation and even the way the feedback influences the derivative effect, in our case a multiplication is used.

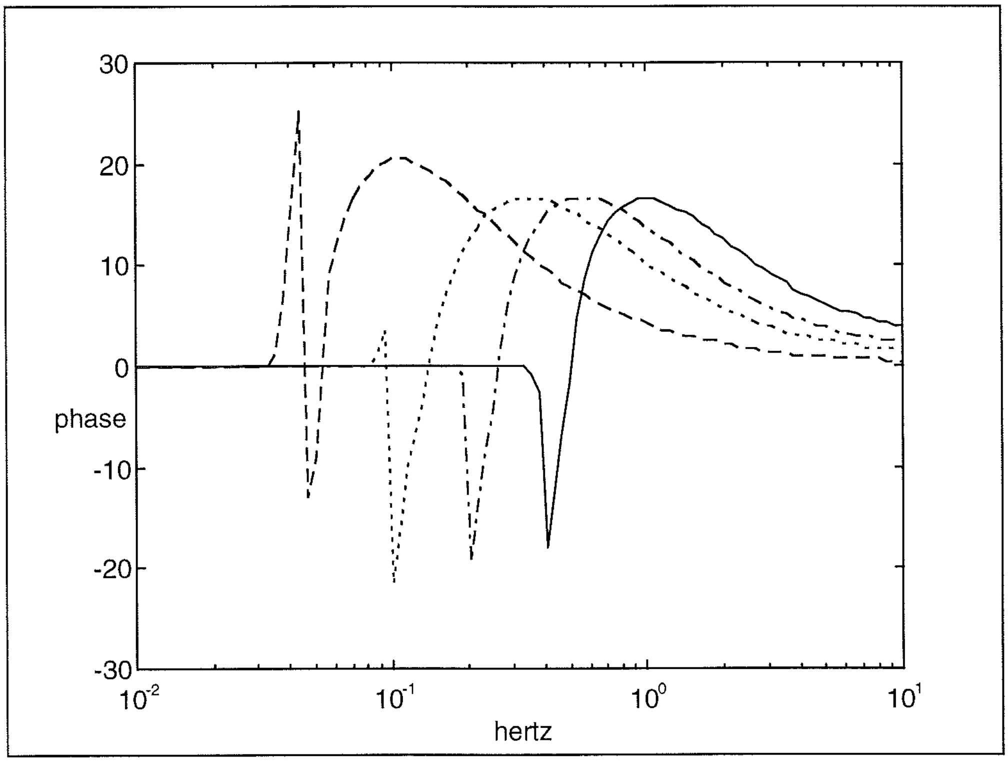


Figure 27
Frequency response (phase loss in degrees) of the modified rate limiter with derivative action.

A weak spot of this rate limiter is of course the fact that the filter uses the derivative of the signal to advance the phase. Generally speaking, using derivatives is not always a positive thing, since the overall filter may be sensible to noise and it may happen that particular limited inputs can cause diverging outputs.
This will be better discussed later on after the discussion of the next filter.

3.8 Phase compensation using derivative and integral action

This filter is a modification of a very basic idea of rate limiting. If we want to limit the rate of a signal, one very spontaneous actions is of course to derive the signal, to limit the value of the derive and then to integrate again the resulting signal. Therefore with a simple chain of three blocks (differentiation, saturation, integration) we are sure to obtain a rate limiter signal.
At the same time a signal passing through such a filter, clearly moves always in the same direction of the movement of the input signal. This is because the input signal's derivative has always the same sign of the output's derivative. In terms of phase lag, this means theoretical zero phase lag.
The reason why such a filter can not be used is because the lack of information due to the signal partially cut by the saturation makes the output signal to drift from the value of the input signal. For the filter's functioning scheme, it makes no difference if the input value is 1000 and the output value is 1, it will only make importance the value of the derivative.
To partially correct this disadvantage, as shown in the next figure, it is possible to introduce some kind of feedback that slowly makes the output signal to drift towards the

desired value. In this way, on the long run, the output signal will always tend to converge back to the same value of the input signal.

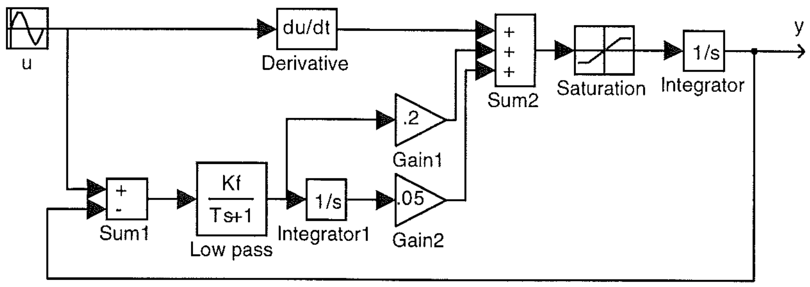


Figure 28
Rate limiter using differentiation-integration

At the same time an integral action is also introduced to correct permanent asymmetries in the derivatives of the signal that would cause a constant error in a proportional-only controller.

At the end, the overall correction circuit results into a PI controller that uses the error between the input and the output to correct the direction of the output.

A low pass filter is used to make the system susceptible only to long-lasting situations and not to sudden changes of the input signal.

The next figure shows a time simulation of the filter. As expected the reversing of the signal is almost instantaneous and there is no phase lag.

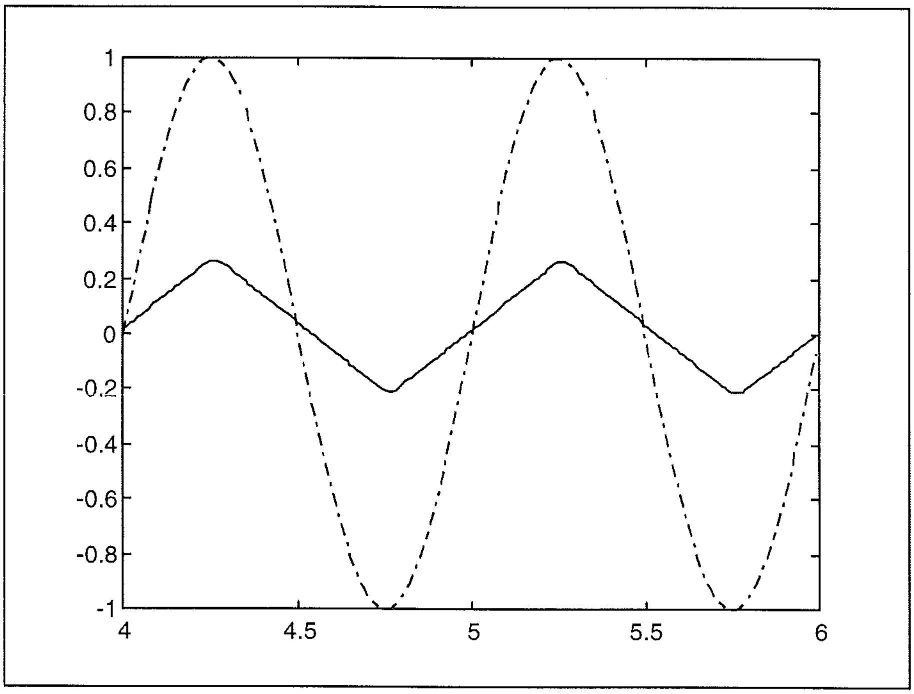


Figure 29
Example of a time response of a rate limiter using differentiation-integration

It is worth to notice that the peaks of the output signal are not sharp but instead present a round shape, due to the fact that when the sinus wave reaches the peak, the signal's

derivative no more exceed the rate limit, therefore the signal is just unchanged and the output has the same shape of the input.
Analyzing as usual the frequency response of the filter, we can observe that the gain diagram is as usual very similar to the others, with little irregular patterns that make it only slightly different than the basic one.

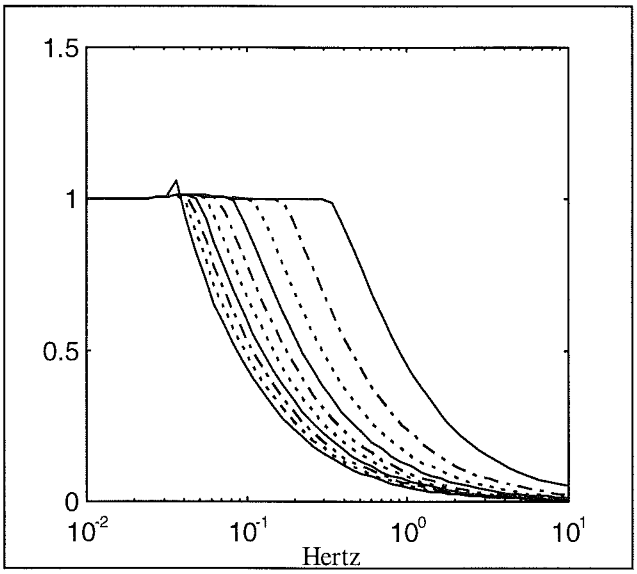


Figure 30
Frequency response (gain) of a rate limiter using a differentiation-integration filter

the phase lag, instead is almost zero at all the frequencies and presents a small gap in phase compensation in a given range of frequencies.
This gap is of course a cause of the presence of the PI correcting system.

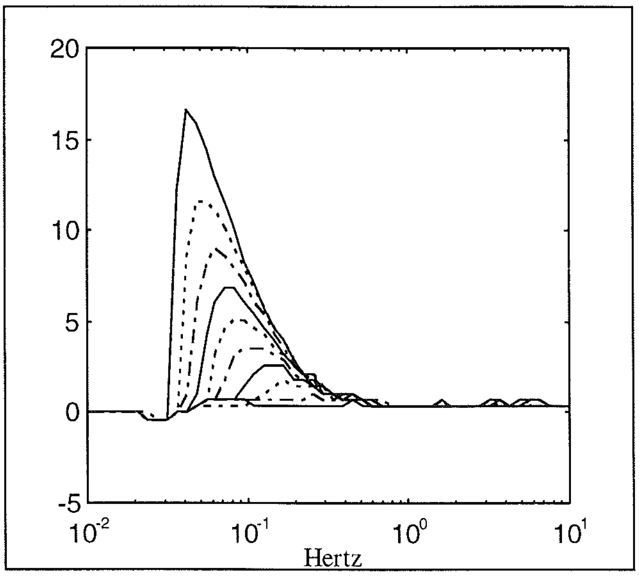


Figure 31
Frequency response (phase lag expressed in degrees) of a rate limiter using a differentiation-integration filter.

The higher the gains of the proportional and integral part, the bigger the phase loss. The integral part is particularly dangerous, since may cause even overshoots of the drifting signal, which result into enormous phase lags.
Now we will investigate how the performance of the filter changes, when parameters are changed. As already said, we have two parameters in the PI controller that can be freely changed.
Generally speaking, increasing the value of these two gains improves the ability of the system to reject any asymmetry of the signal's derivative and at the same time increases the phase loss and the frequency range in which this phase loss occurs.

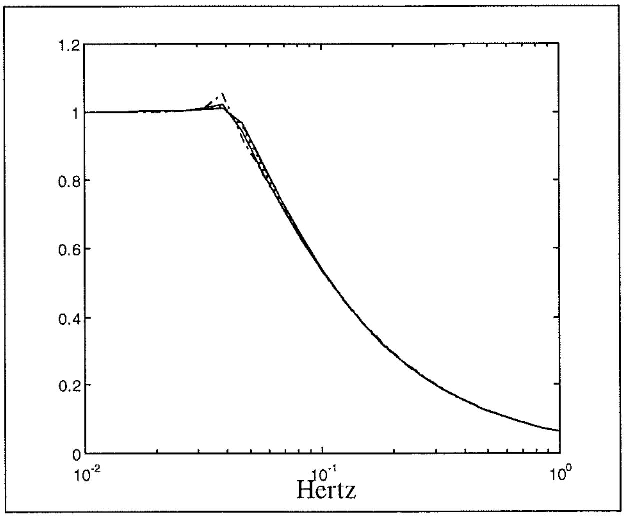


Figure 32
Frequency response (gain) of a rate limiter using differentiation-integration keeping an input amplitude of 4 and varying the feedback time constant (see the block diagram)

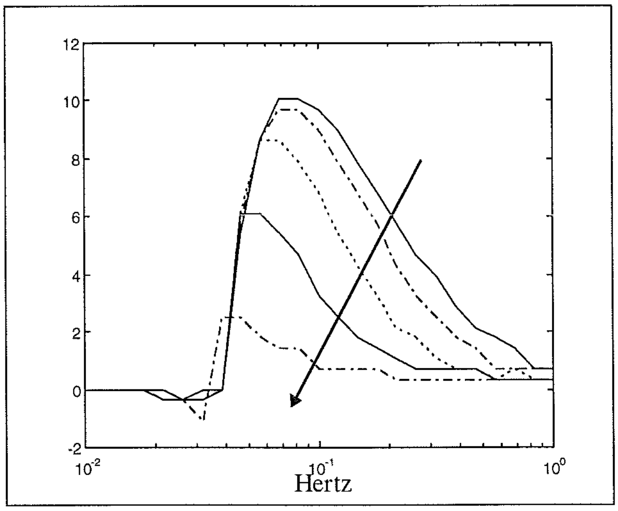


Figure 33
Frequency response (phase) of a rate limiter using differentiation-integration keeping an input amplitude of 4 and varying the feedback time constant (see the block diagram)

Therefore the regulation of the PI controller can not be made without taking into account that improvements in one direction can cause loss of performance in the other direction.
The following two figures, show instead what happens if the time constant of the low-pass filter is changed.
Again we have a situation in which moving towards a certain value of the time constant we have an improvement in the rejection of asymmetric signals and at the same time a higher phase lag and vice-versa.
In this paragraph we have many times said that a rate limiter that uses differentiation as a way to compensate the phase of a signal, is subject to certain dangers.
Generally speaking it was said that asymmetric signals may cause these filters to settle a set point far-away from the real mean value of the input signal, or, even worse, they can cause diverging outputs.
The PI adjustment of the filter was meant also to counteract such asymmetries and we will now analyze the time response to the worst asymmetric signal that can occur, a saw-tooth wave.

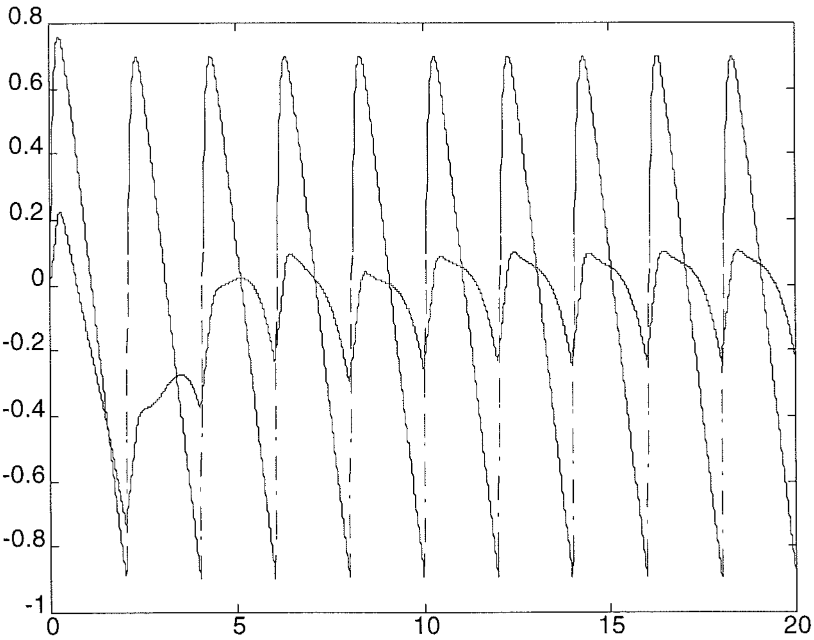


Figure 34
A bad input signal, which has negative derivative over long intervals, may lead to undesirable responses that are corrected by the integral action of the adjusting signal only after some time.

Such signal has derivatives in only one direction and a non-corrected derivative-integrative filter would give as an output a constantly descending signal.
In this case the proportional correction makes the output signal to stop at a certain value, that in general differs from the set point and the integral action slowly corrects the error bringing back the signal to the right level of output after a certain number of periods.
As clearly visible in the picture, not only during the transient the answer is very poor, but also after that steady state is reached, the phase compensation is almost non existent and the overall signal shape is really bad.

The amount of periods needed by the controller to bring the signal to an acceptable level depends as already said by parameters such as the time constant and the gains of the PI controller.
Unfortunately, as previously said, the tuning of these two parameters requires a trade off between different kind of performance requirements.
The integral action is particularly useful to reject this kind of disturbances, but at the same time the overshoots caused by a PI controller with an excess of integral action can lead to a real disaster in terms of phase lag.
If it could be possible to prevent the integral action from causing overshoots still having a high coefficient, we could improve the performance of the filter in terms of asymmetry rejection without loosing performance in terms of phase lag.
The next picture shows an unlucky attempt in this direction :

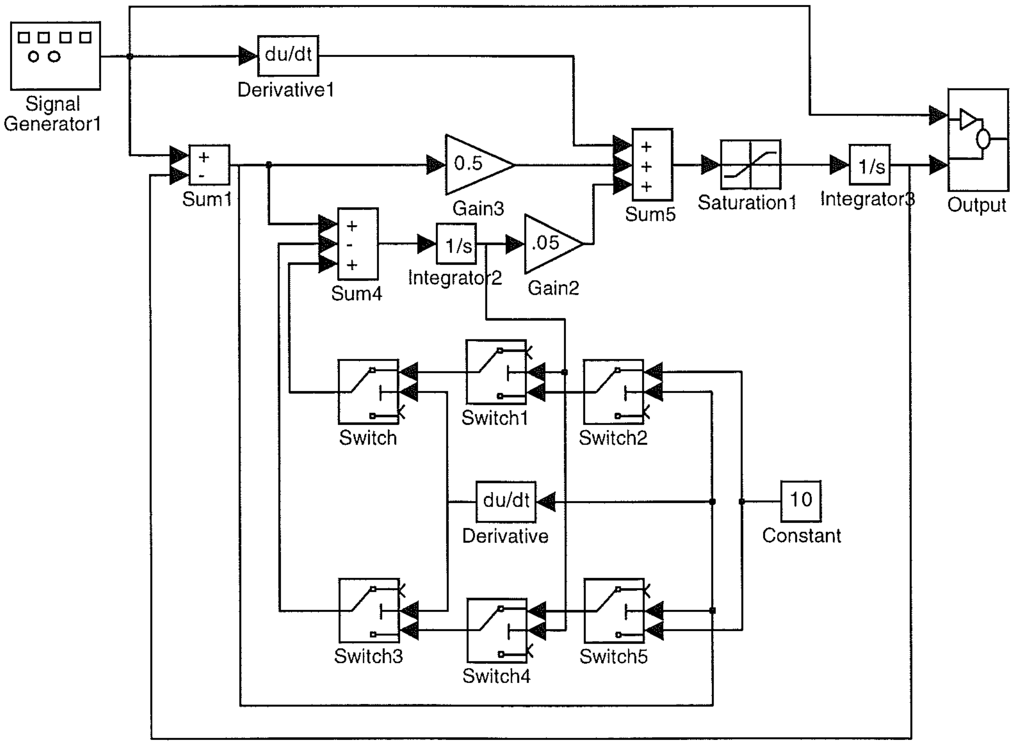


Figure 35
An attempt to avoid overshoots caused by the integral term of the PI adjusting signal in a derivative-integrative rate limiter.

Without explaining one by one the effects of the jungle of switches, the overall circuit is a logic switch meant to detect if the integral action is unproductive.
An integral action is in this case considered unproductive (or even harmful) if the system is going away from the set point and at the same time the integral action is pushing the system further away from the set point.
This can be translated into three condition that must be satisfied in terms of signs of the error, of the derivative of the error and of the integral value.
This kind of logic, introduced in chapter two, tested on a standard PI controller proved to work properly and in most of the cases succeeds in reducing the overshoot, even if not all

the cases are contemplated. (some other times an integral action is not useful and the switch does not turn on since it is not designed to do it.
To understand if such modification can really improve the system, the two filters (rate limiter with differentiation-integration and the same rate limiter with the added integral-killing circuit) are compared in two situations.
In the first one, the filter's PI corrector is tuned with little values while in the second one, the PI corrector bears a higher integral constant.
The following two pictures show the first case. Phase lags for different input amplitudes and frequencies are shown and comparing the two pictures bring up a surprise. The overall performance of the second filter seems worst in terms of phase lag.
Even if the difference it is not that relevant, the presence of the circuit, when not needed, is somewhat harmful to the filter's compensating ability.

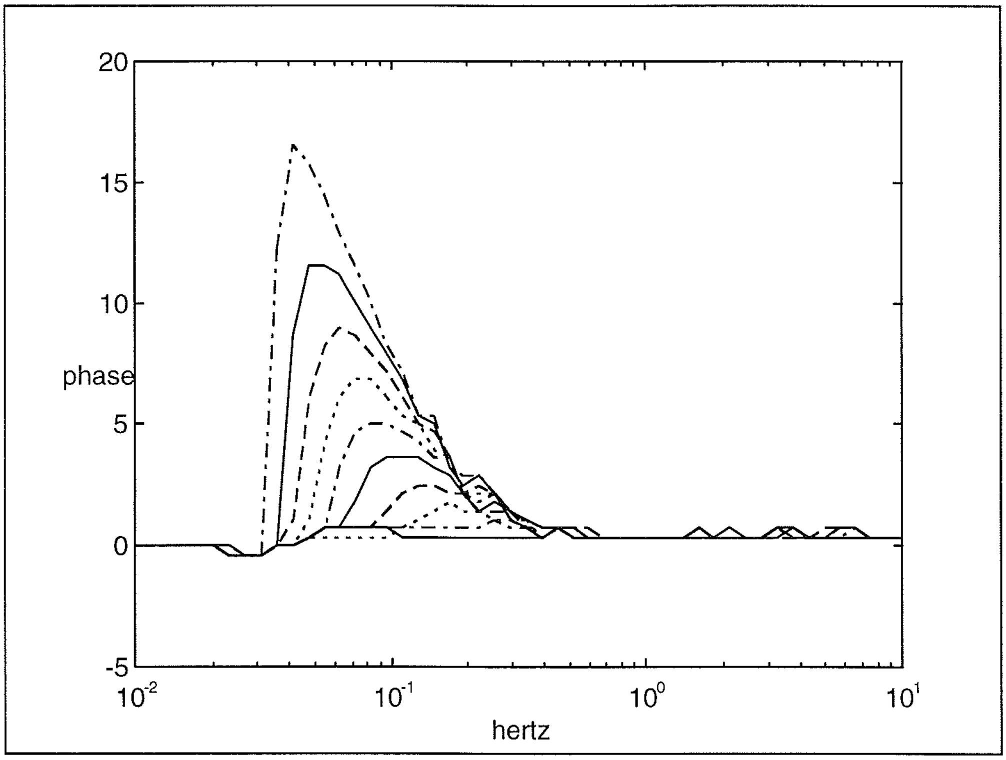


Figure 36
Phase lag of a derivative integrative rate limiter with no integral control.
Case one : small PI constants

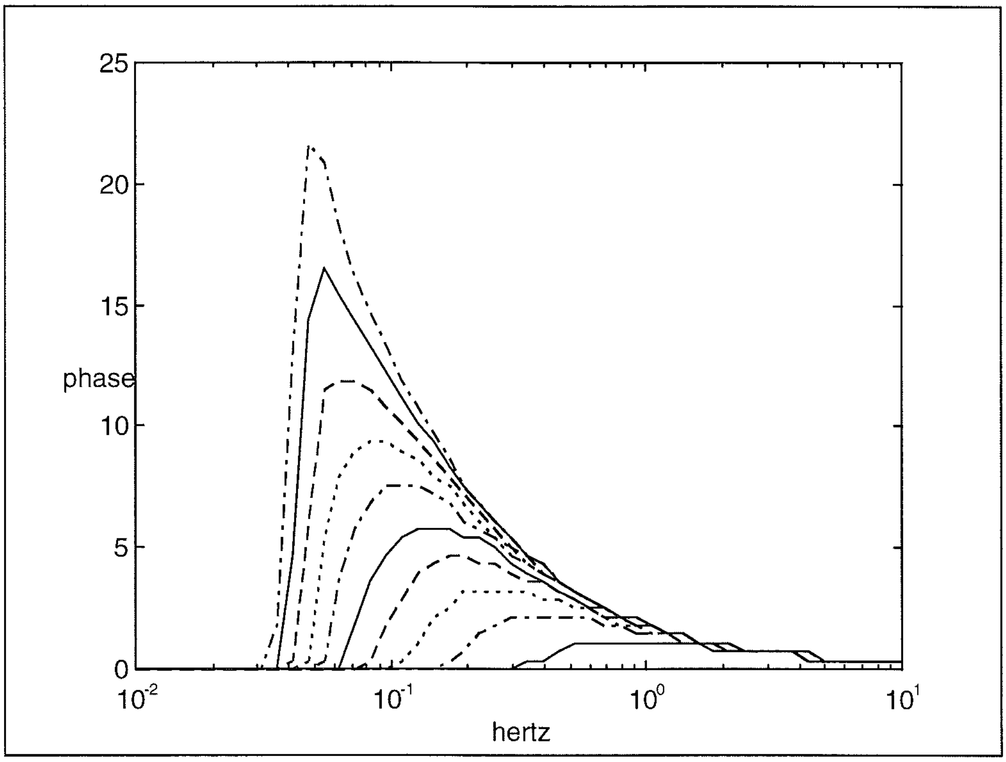


Figure 37
Phase lag of a derivative integrative rate limiter with integral control.
Case one : small PI constants

In the following picture instead the two filters are compared with a higher value of the integral constant and the result is inverted.
This time the first filter presents, with an input amplitude of 10, a catastrophic phase lag of 130 degrees, that is even more than the maximum 90-degrees phase lag that in the worst case can be obtained with a normal rate limiter.
The second filter, instead, thanks to the control on the integral action, presents a phase lag around 45 degrees, which is not small, but it is at least less than half of the previous one.
The question that now arise is if it is really worth to modify to such extent the circuit to introduce control of the integral action. This question becomes more important since it seems that when not needed the control on the integral is even a little bit harmful to the overall phase compensation ability of the filter.
As usual there is no total answer, if the tuning requires high values for the PI controller's constants, then some form of integral control may be necessary to avoid that the integral term takes over and leaf the whole filter to absolute weird responses.

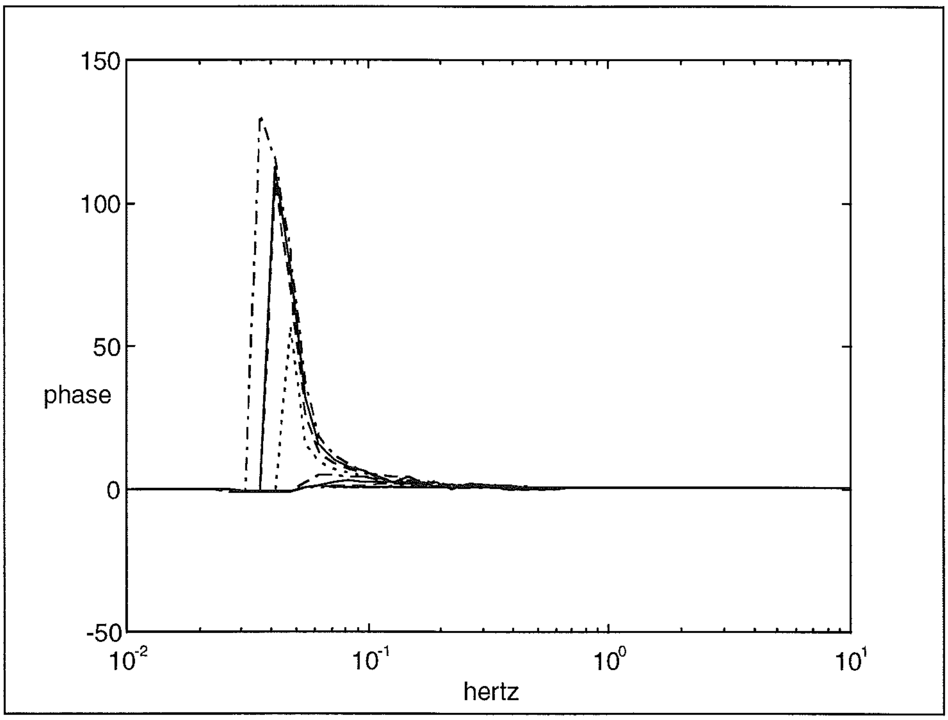


Figure 38
Phase lag of a derivative integrative rate limiter with no integral control.
Case two : large PI constants

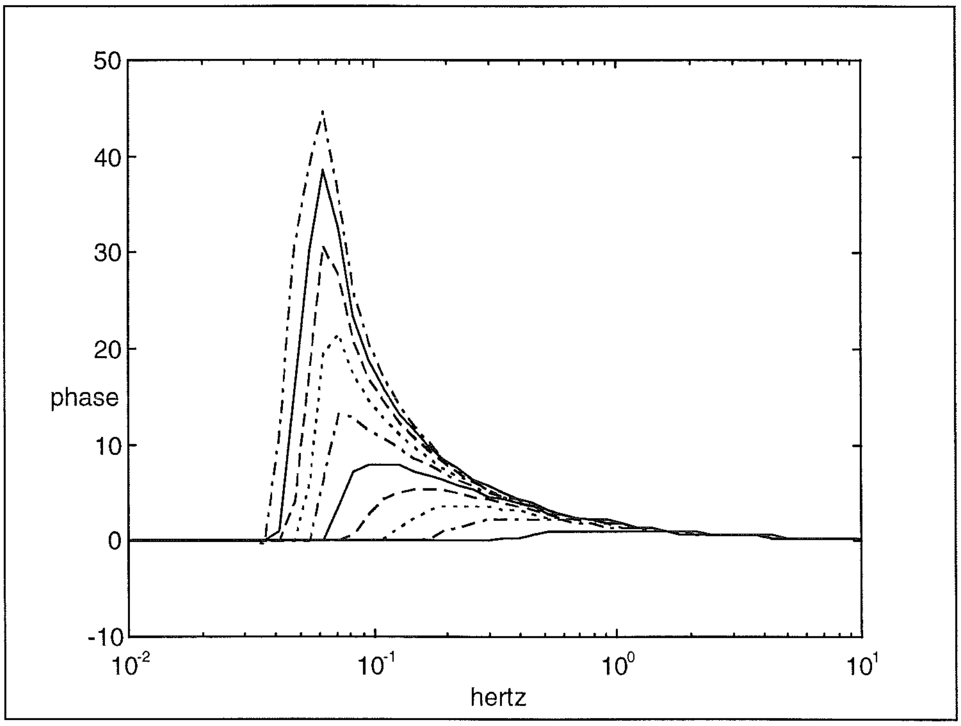


Figure 39
Phase lag of a derivative integrative rate limiter with integral control.
Case two : large PI constants

3.9 Stability analysis of a marginally stable system

In this section we will consider how the different rate limiters will modify the closed-loop stability of a system with a very thin phase margin.
We will consider the third order system described by the open loop transfer function :

$$G(s) = \frac{20 \cdot (s+4)}{(s+1)(s+2)(s+3)}$$

which is characterized by three stable poles and a zero, with a total gain of 40/3.
The system is obviously stable ; if we draw the Bode plot of the transfer function we graphically see the phase margin of the system.
The following pictures show the frequency response of the system, for a wide range of frequencies. Linear analysis give us tools to understand when the closed loop system will be stable.

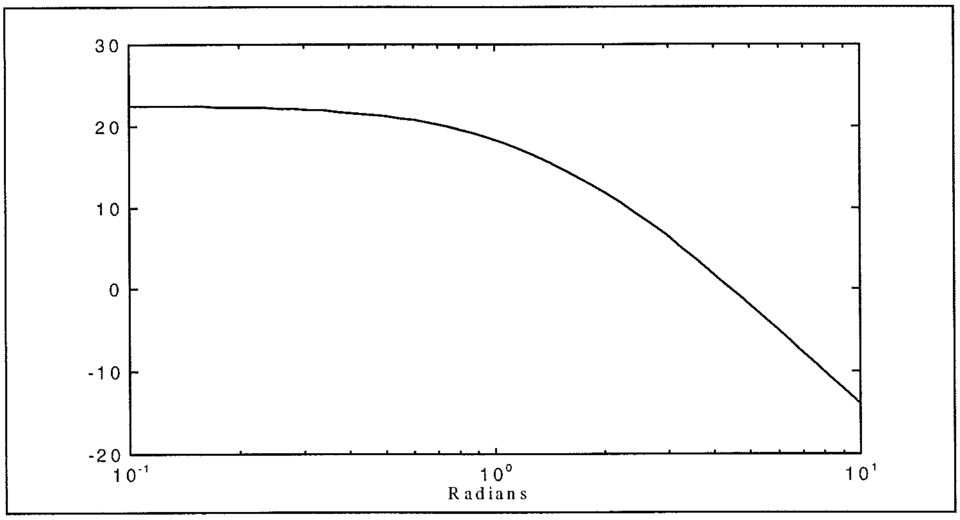


Figure 40
Frequency response of the considered transfer function (magnitude)

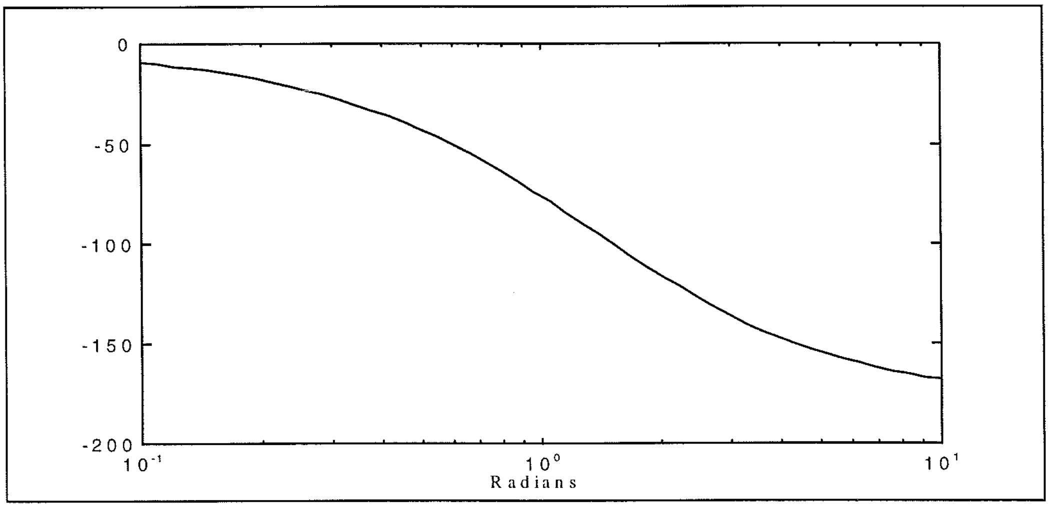


Figure 41
Frequency response of the considered transfer function (phase expressed in degrees)

The next figures instead show the same plant's frequency response, modified with the descriptive function of a rate limiter.
Since the rate limiter is not linear, we don't have a linear transfer function, however we have previously obtained, both analytically and through simulation, the complete transfer function of the non-linearity. Now if we combine frequency response with the transfer function, with a direct multiplication of the two complex number we obtain the frequency response of the overall system for a given amplitude of the input signal.
Even if this has not a general validity is somewhat interesting to observe how the curves are modified by this.

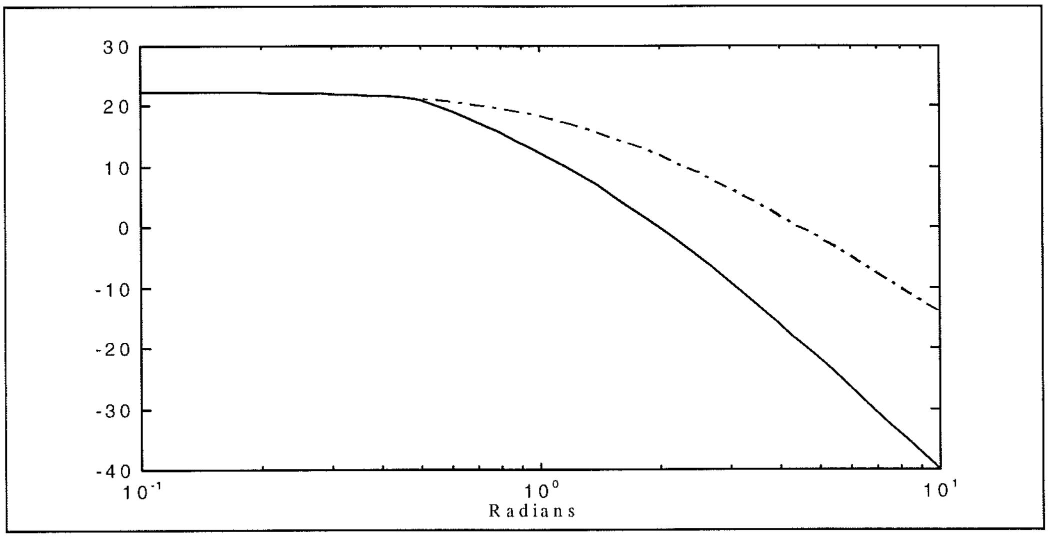


Figure 42
Frequency response (gain) of the transfer function affected by the rate limiter and calculated for an input amplitude of ½ (solid line) compared with the transfer function of the unaffected transfer function

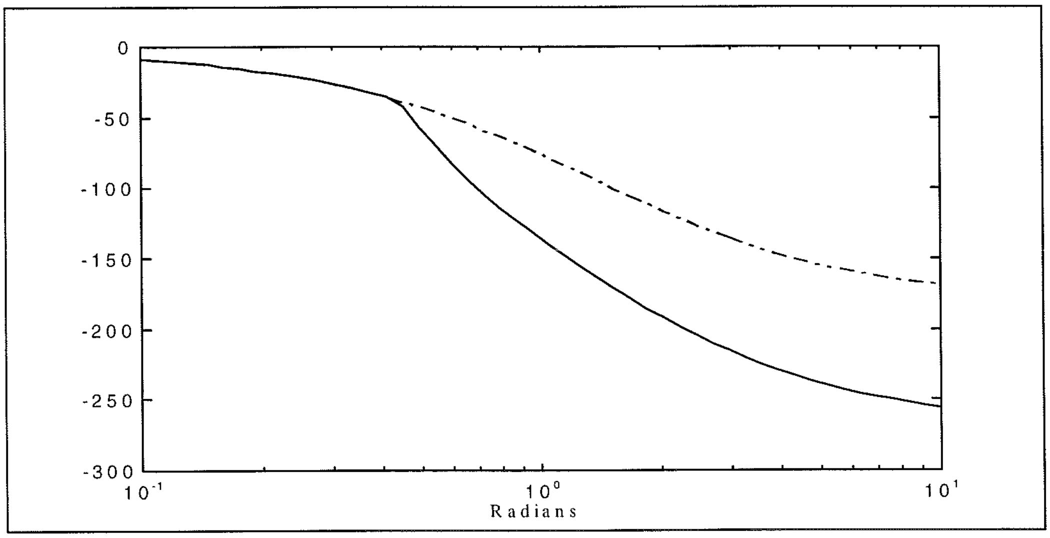


Figure 43
Frequency response (gain) of the transfer function affected by the rate limiter and calculated for an input amplitude of ½ (solid line) compared with the transfer function of the unaffected transfer function

The first thing clearly visible is the dramatic loss of phase margin. This clearly explains why the system described in the first paragraph became unstable to fast inputs, where the rate limit was strongly exceeded.
The considered plant, as the one considered in the first paragraph, is marginally stable. Plotting a Nyqvist diagram clearly shows that the system is not far from the (-1,1) point on the complex plane. If again we apply the same procedure and we combine the linear plant with a rate limiter frequency response, we see clearly, as shown in the next figure, that the stability limit is exceeded, meaning that this time the closed loop is expected to be unstable.

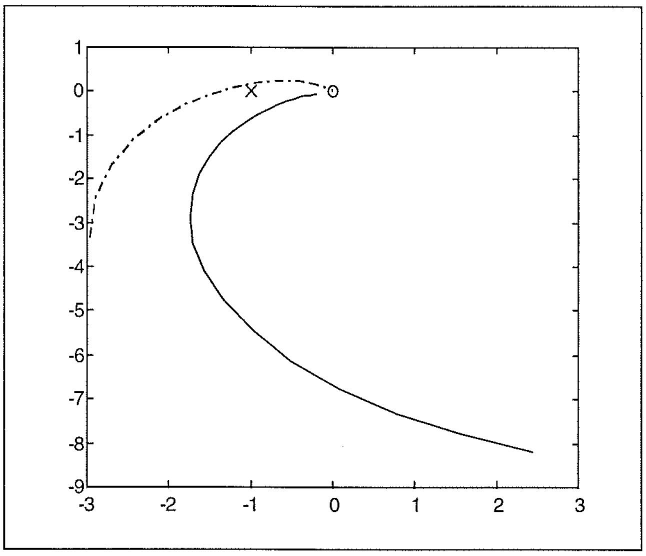


Figure 44
Nyqvist plot of the transfer function of the considered system (solid line) and of the transfer function affected by the rate limit (dashed line)

In the following figures the same plot is made comparing the different rate limiters. All the considered rate limiters succeed in stabilizing the system again.

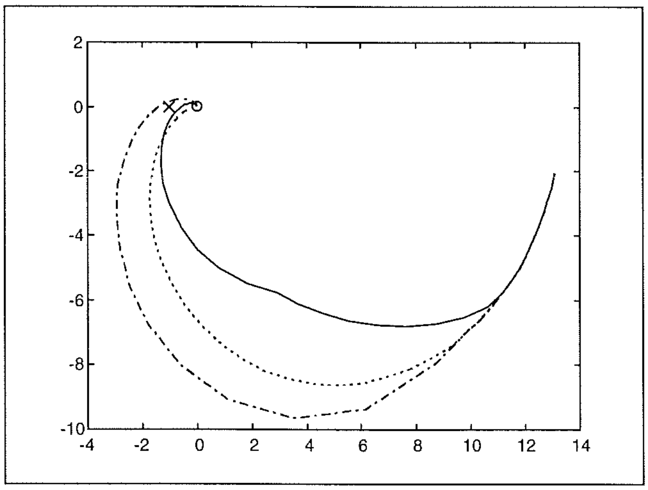


Figure 45
Comparison between the Nyqvist plots of the original transfer function (dotted line), the transfer function affected by the basic rate limiter (dashed line) and the transfer function corrected by the basic filter with feedback (solid line)

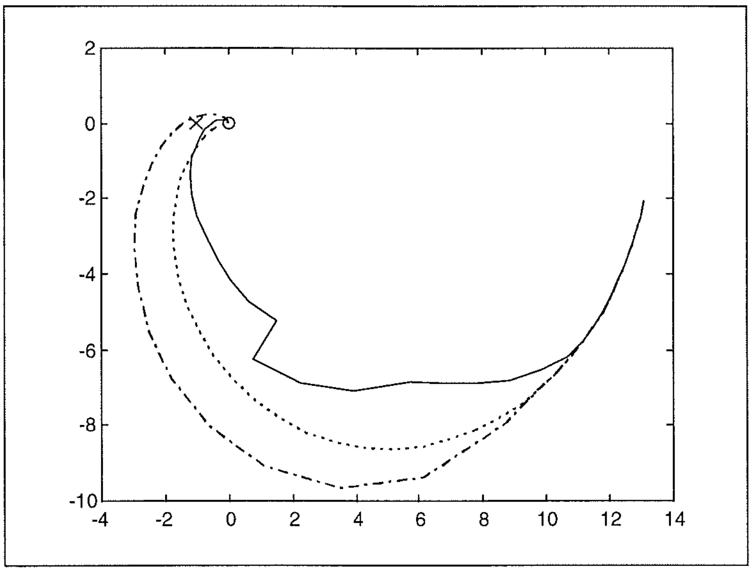


Figure 46
Comparison between the Nyquist plots of the original transfer function (dotted line) the transfer function influenced by the basic rate limiter (dashed line) and the transfer function corrected by the filter with feedback and bypass, computed as usual with an input amplitude of ½ (solid line).

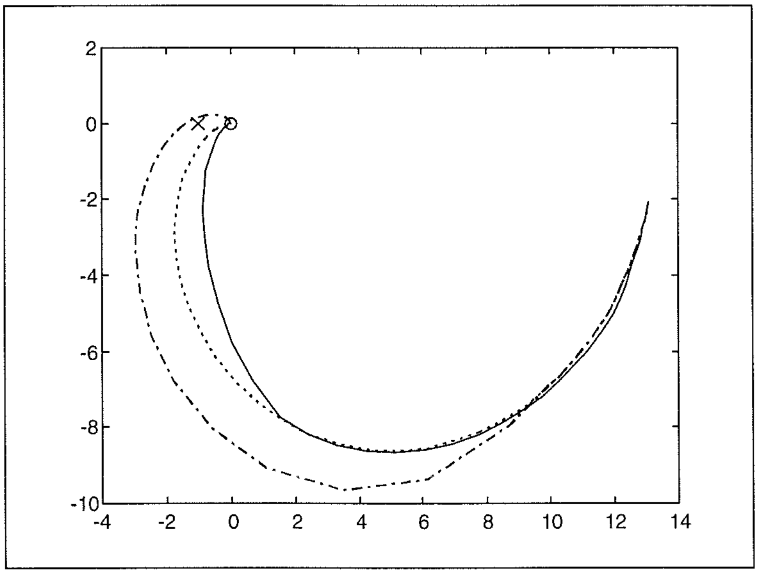


Figure 47
Comparison between the Nyquist plots of the original transfer function (dotted line) the transfer function influenced by the basic rate limiter (dashed line) and the transfer function corrected by the filter with differentiation-integration (solid line).

3.10 Time domain simulations

In this last paragraphs we will test the filters on time-domain simulations analyzing their impact on more complex plants.
As a first case we will consider again the marginally stable system shown at the beginning of the chapter. Time simulation showed that fast inputs could lead the system into instability.
Throughout the chapter it was shown why rate limits reduce stability margins of systems and now the same simulation (see figure 3) is proposed introducing the effect of these filters.
The next four figures show the effect of a filter with feedback, of a filter with feedback and bypass, of a filter with derivative action and of a filter with differentiation-integration.
The four filters manage to stabilize the system and on the steady state the output tends to the input value.

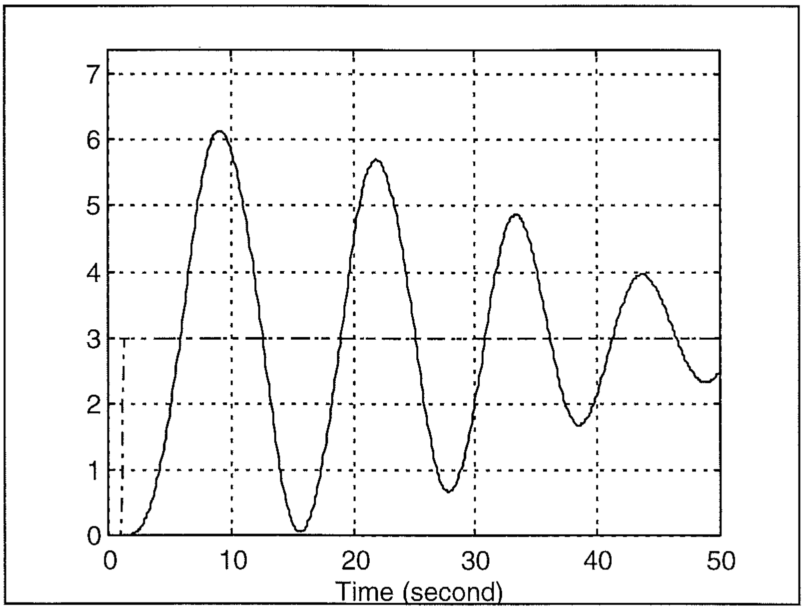


Figure 48
Time response of the plant of figure 3 (solid line) to a sudden input (dashed line), under the effect of a rate limiter with feedback

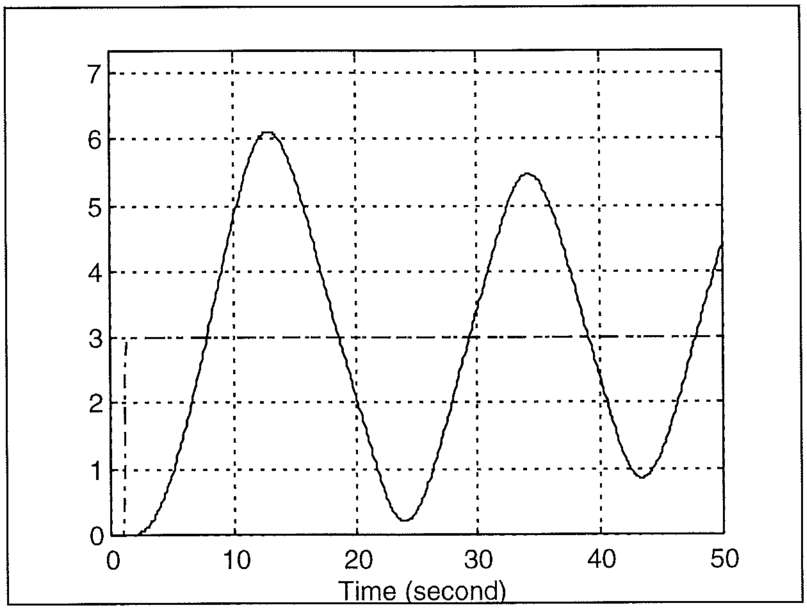


Figure 49
Time response of the plant of figure 3 (solid line) to a sudden input (dashed line), under the effect of a rate limiter with feedback and bypass.

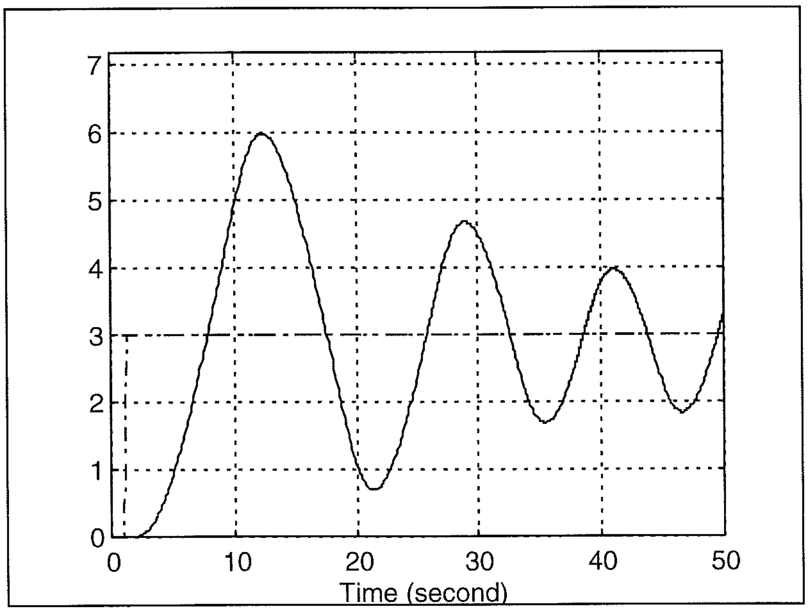


Figure 50
Time response of the plant of figure 3 (solid line) to a sudden input (dashed line), under the effect of a rate limiter with derivative action.

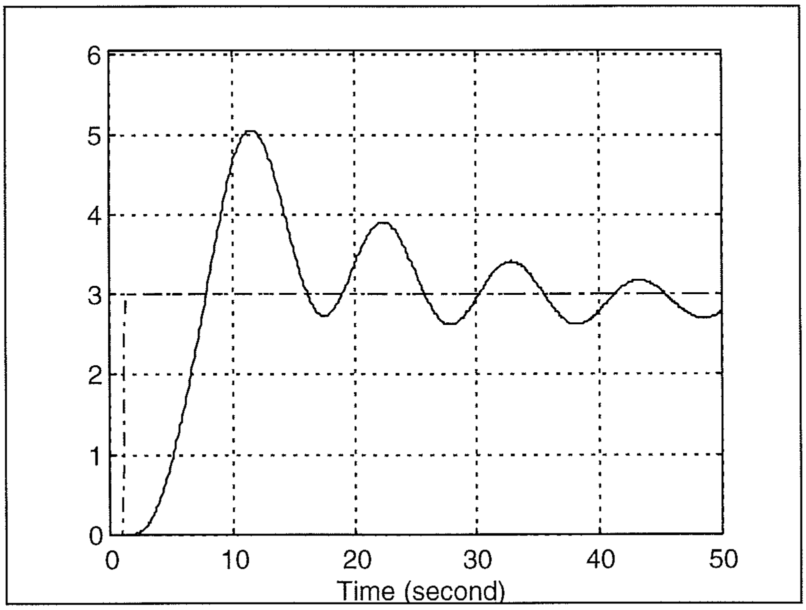


Figure 51
Time response of the plant of figure 3 (solid line) to a sudden input (dashed line), under the effect of a rate limiter with differentiation-integration.

As a final test we will show the same kind of time simulation with a different system. This time the fourth order plant is marginally stable itself and the loop is closed without an additional controller.
This will test the filters with a different plant and at the same time will the limit of a derivative-integrative filter.

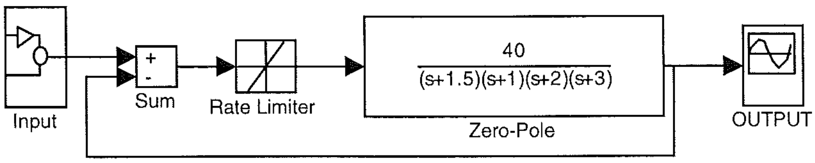


Figure 52
Another marginally stable system without a controller before the rate limiter.

The next figure shows, as done before the time response of the considered system with a rate limiter with feedback.
The system manages to stabilize again the plant and everything goes in the same way as before.
Notice that this time the set point is not reached by any of the four systems since there is not anymore the controller with the integral action.

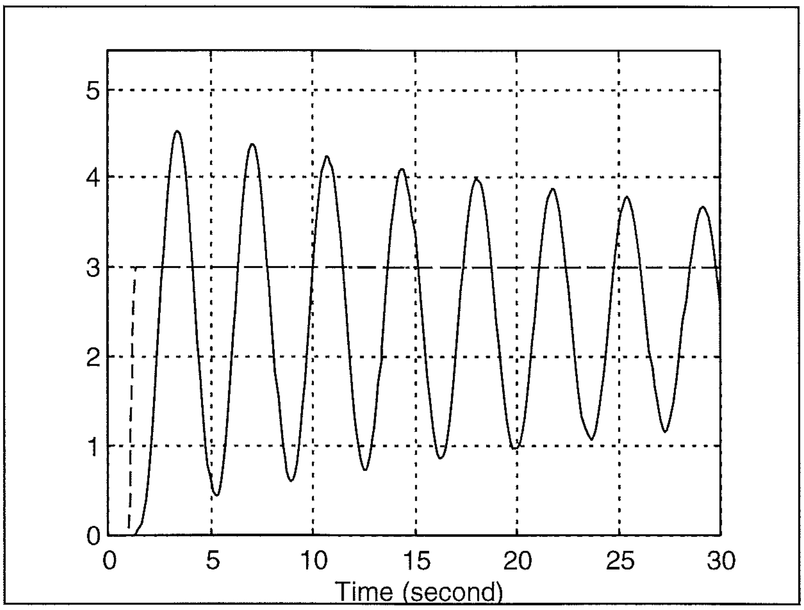


Figure 53
Time response of a marginally stable system with a rate limiter with feedback in the control loop.

The next figure shows what happens if the rate limiter with feedback and bypass is not correctly tuned.

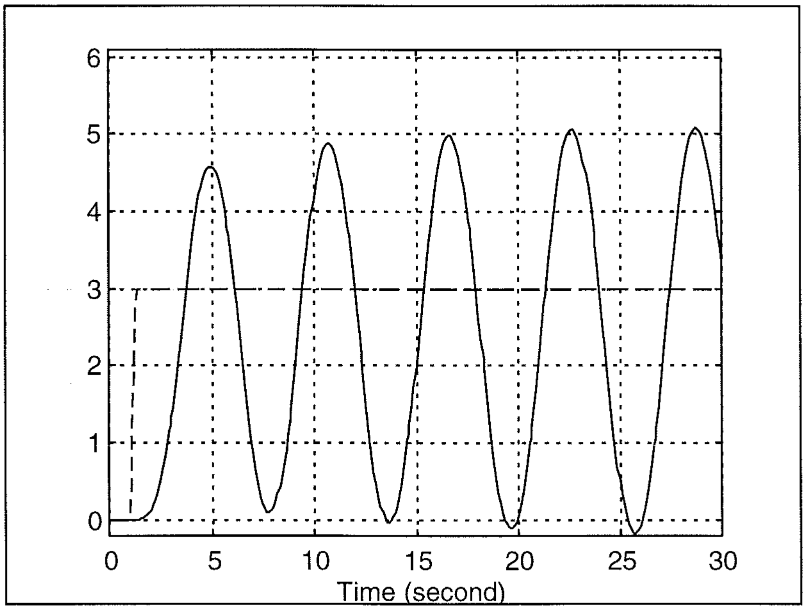


Figure 54
Time response of a marginally stable system introducing a rate limiter with feedback and bypass in the control loop with improper settings.

The presence of a low pass filter allows two degrees of freedom to customize the effect of the filter. If these parameters are not correctly settled, the filter may fail to provide enough phase compensation when needed.

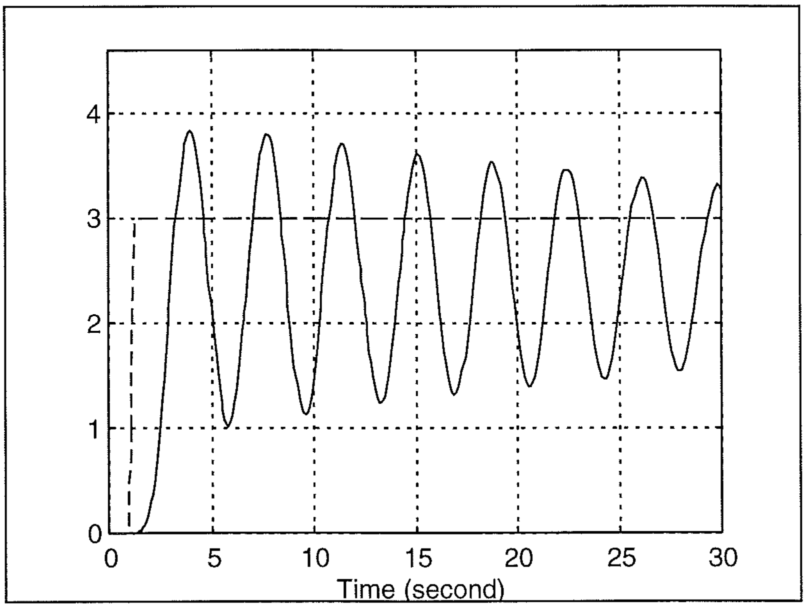


Figure 55
Time response of a marginally stable system introducing a rate limiter with feedback and by-pass in the control loop.

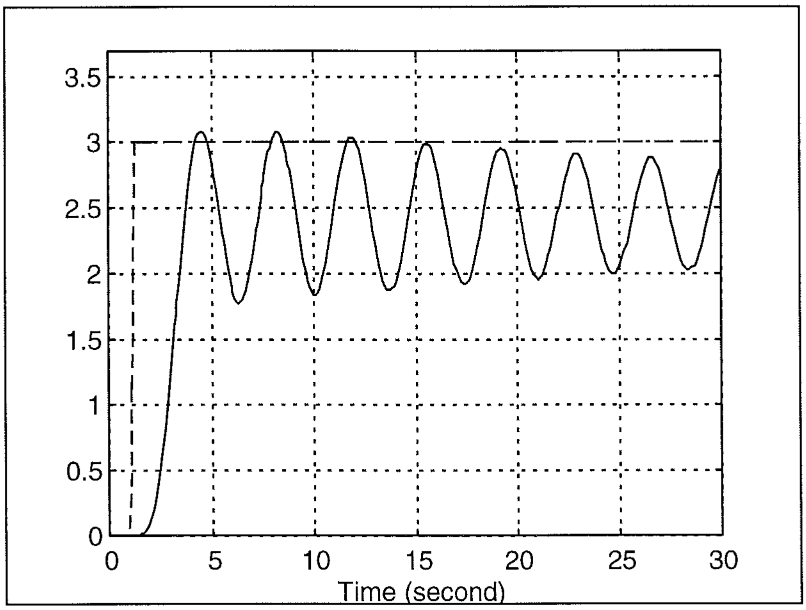


Figure 56
Time response of a marginally stable system introducing a rate limiter with derivative action in the control loop with improper settings.

In the next two figures, a rate limiter with derivative-integrative effect is used to compensate the phase loss in the control loop.
The performance of the filter this time is less satisfactory than in the previous case. The reason for this is that this time there is no controller between the input and the rate limiter, therefore the whole input step is fed directly to the rate limiter.
As already known, this rate limiter reacts very poorly to step inputs and therefore the output rises up only due to the PI correction.

The next figure show a case with small values in the PI constants, while in figure 58 is shown a case with higher values in these constants which result into an undesirable overshoot.

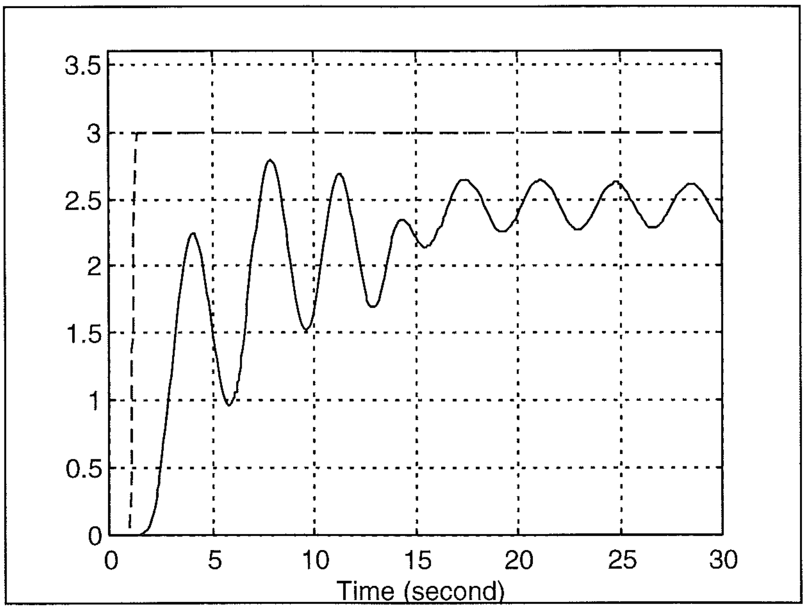


Figure 57

Time response of a marginally stable system introducing a derivative-integrative rate limiter in the control loop with small values of PI parameters.

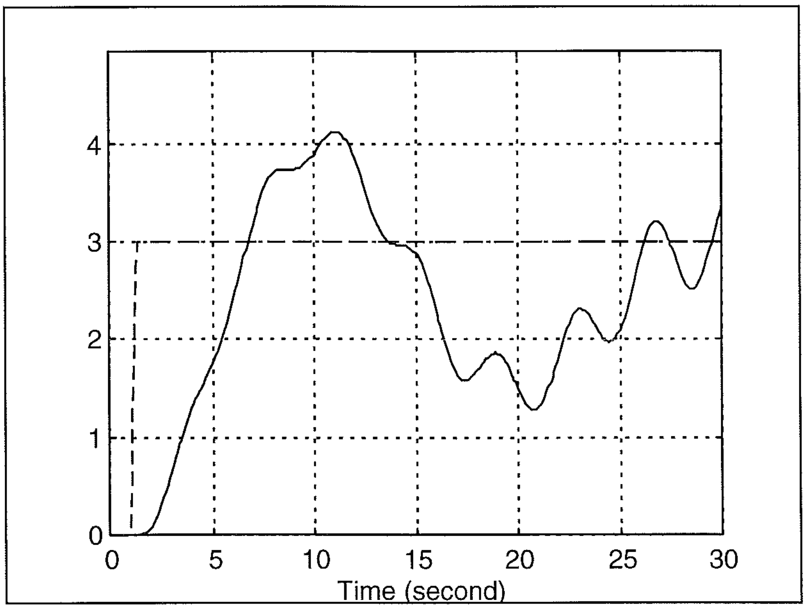


Figure 58
Time response of a marginally stable system introducing a derivative-integrative rate limiter in the control loop with larger values of PI parameters.

3.11 Conclusions

Different compensating methods were tested and they showed different advantages and disadvantages.
The approach based on differentiation and integration presents a problem of loss of information. It is more sensitive to noise, since it is highly non linear there may be instabilities for certain inputs. With the PI adjusting controller the filter is more reliable and able to reject bad signals, its response to signal with large derivatives is poor. A low-pass filter before the rate-compensator may therefore be required.
Increasing the influence of the adjusting controller too much improves the rejection of the bad signals but at the same time decreases the performance of the compensating action. Therefore it is necessary to make trade-offs for specific applications.
The differentiation-integration approach has the benefit of a theoretically zero phase-lag. In practice there will be a small phase lag due to the effect of the adjusting feedback and a possible low-pass filter.
The filter with derivative action reacts very good in simulations and provides usually very good phase compensation. It also gives a satisfactory response to step inputs. Unfortunately the filter is almost completely defenseless against signals whose derivative is highly asymmetrical. and in addition it is susceptible to noise and disturbances. These drawbacks may be alleviated with an improved filter.
Phase compensation through feedback leads to a filter that has a quite smooth, regular response to different signals. The advantage of this filter is that it is not based on the derivative of the signal. Therefore it does not suffer from the typical disadvantages associated with this approach. This filter has been tested in real situation and showed good handling characteristics, giving the pilots a good feeling of the response. It is is now implemented in the Flight Control System Software of the fighter JAS-39 'Gripen' both on Longitudinal and lateral dynamics.
The filter only compensates certain frequencies. It just behaves like a traditional limiter without any phase compensation to high frequencies. When the limited bandwidth is enough for a system to avoid potential instability the filter based on feedback is preferred due to it is reliability ; this happens in most cases since high frequency phase-lag very rarely endangers stability. If on the other hand the need of a higher bandwidth is felt, then it is necessary to change filter, since it does not seem possible to extend the bandwidth of the filter with feedback with an acceptable freedom.
Looking for a better phase compensation may lead eventually to other filters that, as shown, bear other undesirable characteristics.
Looking at the problem from a general point of view we can observe that phase compensating is a non-causal problem, and it seems a never ending run to pursue a perfectly compensated filter that is not susceptible to particular inputs that fool its logic. The reason for this is clearly understandable considering that if we try to design the perfect answer to two kind of signals very different from each other, we realize that it is not possible to know how the filter should behave without knowing the future behavior of the signal.

4. A first order unstable system

4.1 Introduction

We will now investigate the effects of saturation and rate limitation on an unstable system. Consider a very basic 1st order system, whose transfer function is given by :

$$G(s) = \frac{1}{s-\lambda}$$

with λ as a positive real number.
Obviously the unique pole of the system is λ itself and therefore the system is unstable.
A way to stabilize the system is to use a feedback equal to k where k is a real positive constant greater than λ.
The system's state space representation would be :

$$\dot{x} = \lambda x - kx + u$$

resulting in a negative real pole that ensure stability.

4.2 Input saturation

If we consider the system's input limited by a saturation to +/- u̲ then it is easy to see that if we let the state variable x to go beyond the value of u̲/λ we are no more able to control the evolution of the system which starts to diverge towards positive or negative infinity.

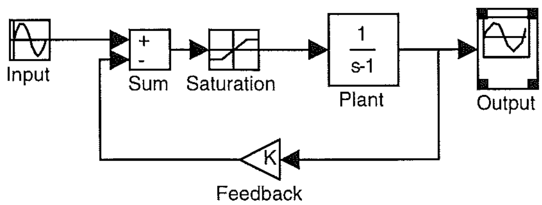


Figure 1
A first order unstable plant controlled with a limited signal

The input of the system can be for example a manual control over the status variable and it is evident that improper inputs can drive the system into instability.
It is necessary therefore to design a controller that, getting close to the critical limit, prevents the manual control to give inputs that drive the system towards instability.
Since the plant is a first order system, a first immediate solution can be to limit the external input with another saturation, so that it will not possible for the operator to exceed the limits given by the operating envelope.
In the following examples, unless differently specified we will assume numerical values of the parameters as follows :

L	1

K	2
Saturation +/-	1

These changes will result in the following modified block diagram :

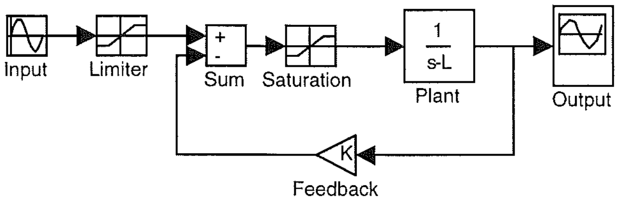


Figure 2
A saturation of the control output ensure that a first order system keeps within given boundaries if no disturbances are applied

The limiter prevents the absolute value of the input from exceeding the value of 0.9 (cautelative), and therefore as a response to large inputs, the system gently approaches the defined limit with a first order dynamic (its own artificial dynamic, given by the feedback). In the next figure a time response to a sinusoidal input of amplitude equal to 2 :

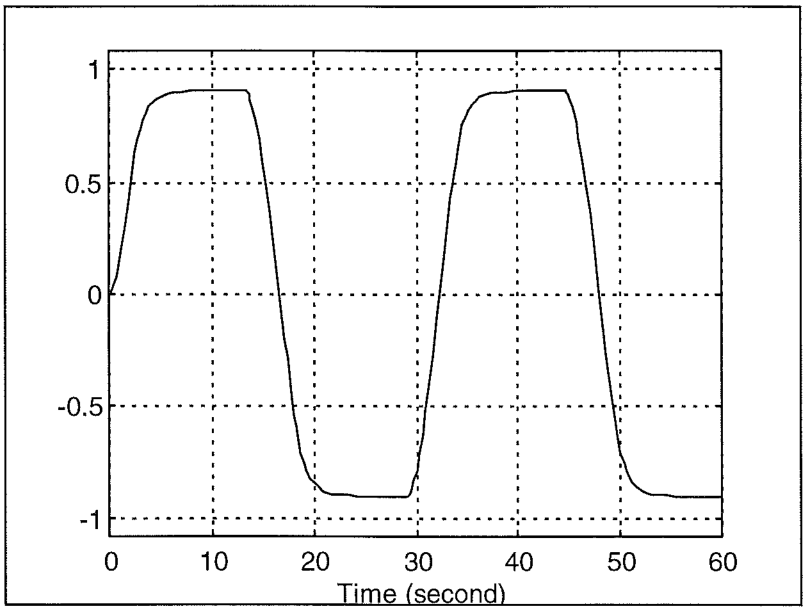


Figure 3
Answer of the previous system to a sinusoidal input with an amplitude of two.

This method assures that a first order unstable system evolves safely, but at the same time it impairs performance, since it is not really necessary to limit the input unless the system is really reaching the boundary of the controllable zone.
A better way can be to dynamically restrict the available input, depending on the status of the system.

A very simple example of this technique can be realized getting the feedback from the output and adding it to the system's input just before the safety saturation (which has to be increased) and subtracting it immediately after.
In the example that follows the output is multiplied by two, therefore the signal will range from -2 to +2 within the controllability range. This signal is added to the input and the whole signal passes through a saturation which has a maximum absolute value of 2.9.
After the passage through the saturation, the feedback signal is subtracted again and the remaining input is fed into the system as usual.
This is how the block diagram will look like :

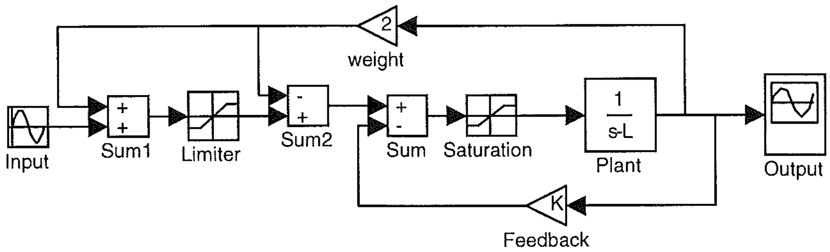


Figure 4
A modification to the saturation of the control input to allow larger control inputs when the controllability of the system is not treated

In this way, the input signal is restricted up to different maximum values depending on the system's state ; if the system's state variable is zero, then the input will be restricted to +/- 2.9 while if the state variable is at the edge of the controllability zone, then the maximum absolute value allowed will be 0.9, resulting in a recovery action that will bring the system towards the controllable zone again.
Of course the values can be tuned acting on the feedback constant and on the value of the safety saturation, resulting in different safety threshold and different dynamics of the system. The more importance is given to the feedback signal compared to the input signal and less smoothly the system will stop when approaching the edge of the controllable zone.

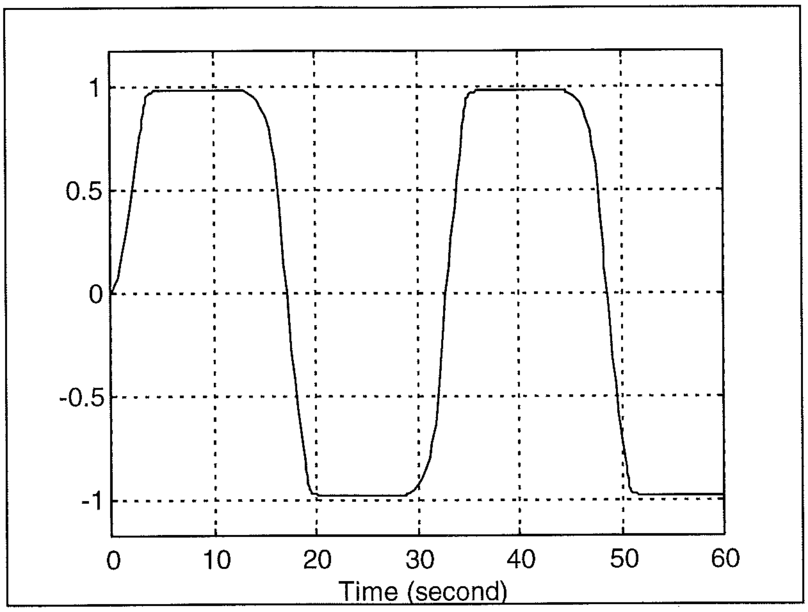


Figure 5
The same simulation performed with the modified block diagram

As a last solution we may think a diagram that leaves the operator free to control the system with any range of input and stops the control action only when the system is really going towards a dangerous value of the state variable, lets say, in our example, that the system has to be stopped when the state value has an absolute value greater than 0.9, and the system has to be driven back to value of 0.9.
First order systems can be immediately stopped and therefore the block diagram doing this has a very simple logic :

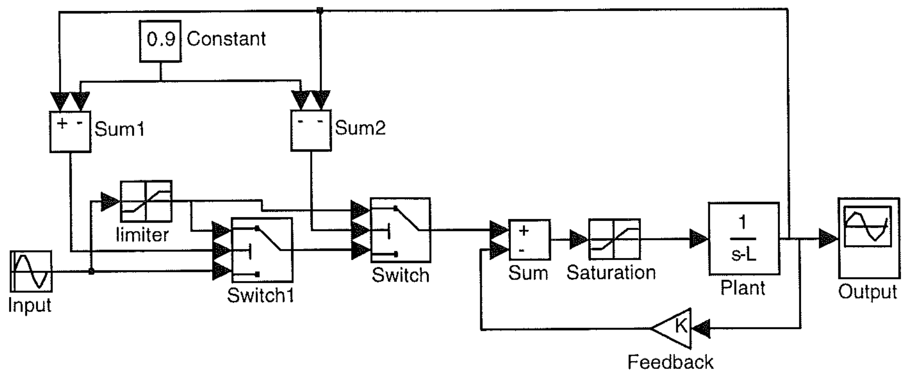


Figure 6
Safety switches may take control of the input when the state variable exceeds certain values

In this case the input is limited between -0.9 and +0.9 only if the status variable exceeds the absolute value of 0.9, otherwise any input is allowed and fed to the plant.
The time response is different. In this case the system is abruptly stopped only at the threshold and in the path it is evident the presence of crisp edges in the trajectory.
This is a time response to a sinusoidal input with a magnitude of 2 and bearing the same frequency of the previous examples :

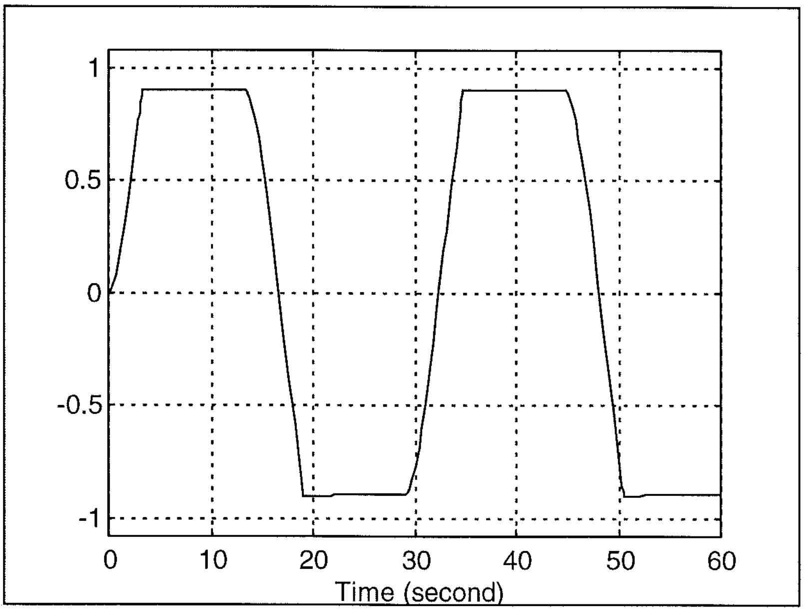


Figure 7
Simulating the system with safety switches

Now instead of proceeding with standard blocks we can think of adding a programmable controller that takes the external input as well as the feedback signal and evaluates in real time the most appropriate signal according to the situation and the control strategy chosen. The previous block diagram can be modified into the following way :

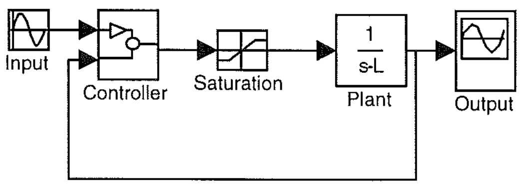


Figure 8
A programmable controller may take the place of all the other block modifications. The software can be changed at will without acting on the rest of the system.

and a programmable block was added to be able to introduce an arbitrary logic between the input signals and the final system's input.
In a real situation, the controller block can be a standard programmable Digital Signal Processor that let us change at will the control law without changing the hardware used.
We will now try to program the Controller Block so that it will behave like the system showed in figure 6.
To fulfill this aim the controller behaves differently according to three different system's states :

$-p < x < +p$	• This is the 'normal' operating situation, the controller simply multiply the feedback by K and subtract it from the manual input to obtain the system's input
$x < -p$	• A negative 'out of range' is occurring, the controller raises up to -p any external input smaller than -p
$x > p$	• In this case, the controller cuts down to p any external input greater than p

It is evident that whenever the system approaches the +p or -p status then it is not possible anymore to give a control input greater than $|p|$ and therefore from the system's state space equation we can see that whenever $x = p$ the resulting derivative can only be negative or zero.

If we set p to be less than $\underline{u}/\lambda$ we can assure that no matter the external input given, the system remains stable.

Here follows an numerical simulation showing the system under a sinusoidal input that goes beyond the range given to the controller.

As for the previous examples, the value of p is set to 0.9.

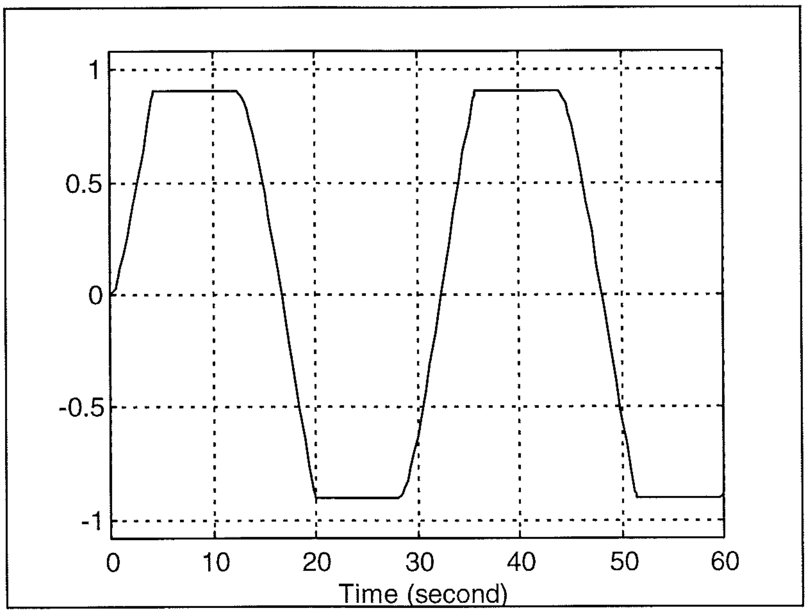


Figure 9
Implementing safety switches in the programmable controller leads to the same results obtained with block diagram modifications

4.3 Input rate limitation

The simplicity of the controller used till now bases its principle on the fact that in a 1st order system we have direct and instantaneous control of the speed at which the system is changing state.

If we suppose that the rate at which we can change input is limited :

$$-r < \dot{u} < r$$

it is not possible anymore to guarantee that the previous controller manages to stop well in advance the system preventing it from going into an unstable zone.
Given a certain value u of the control input we have to calculate the point at which a full control effort in the opposite direction is merely enough to prevent the system from going unstable.
For simplicity we will focus now on the positive instability range, since all our calculations can be easily reversed.
Given a constant deceleration effort equal to the maximum rate allowed, the system's differential equation can be written as :

$$\dot{x} = \lambda x - \dot{u} t$$

It can be shown that solutions of this differential equation are the equations represented by :

$$x = c e^{\lambda t} + \frac{\dot{u}}{\lambda} t + \frac{\dot{u}}{\lambda^2}$$

where *c* is an arbitrary constant.
We will now determine the extreme of the function by finding the time at which the 1st derivative of the system is zero.

$$x' = \lambda c e^{\lambda t} + \frac{\dot{u}}{\lambda} = 0$$

Substituting this relation into the original function and defining "M" as the maximum value that the function has to reach yields to :

$$x = -\frac{\dot{u}}{\lambda^2} e^{\lambda\left(t - \frac{\lambda M}{\dot{u}}\right)} + \frac{\dot{u}}{\lambda} t + \frac{\dot{u}}{\lambda^2}$$

Now, changing the time variable we shift the frame of reference so that the function assumes the value M per T=0, we get :

$$x = -\frac{\dot{u}}{\lambda^2} e^{\lambda T} + M + \frac{\dot{u}}{\lambda} T + \frac{\dot{u}}{\lambda^2}$$

With the above relation we are able, given the parameters and the maximum values of input and input rate, to calculate the real value of the state variable beyond which it is impossible to recover the system.
As stated in the previous paragraph, M is equal to u/λ. While the minimum time needed to completely reverse the control input in the worst case is :

$$T_{MAX} = -\frac{2\underline{u}}{\dot{u}}$$

indicated in a negative value to remember that the found equation has meaning if computed for negative values of the time variable.
Substituting this time into the equation gives :

$$x = -\frac{\dot{u}}{\lambda^2} e^{\left(-2\lambda \frac{u}{\dot{u}}\right)} - \frac{u}{\lambda} + \frac{\dot{u}}{\lambda^2}$$

that indicates the value that x can assume for the system to remain stable and controllable.
Plotting the previous equation against the rate limitation as independent variable with a real unitary pole and unitary input limit we get the following diagram which suggests the boundary ensuring the controllability of the system under any input.
As one could expect the curve reach asymptotically the unitary value if we increase indefinitely the rate limit ; at the same time a very low value of the rate limit can give a negative result. It is clear that under this condition it is impossible to guarantee the controllability of the system.
Drawing up conclusions, for very high values of the rate limit this controller is good enough as it ensure the controllability of the system without penalizing too much the system performance, while for low values it becomes necessary to use a controller that takes into account the current value of the input value to change dynamically the safety boundaries.

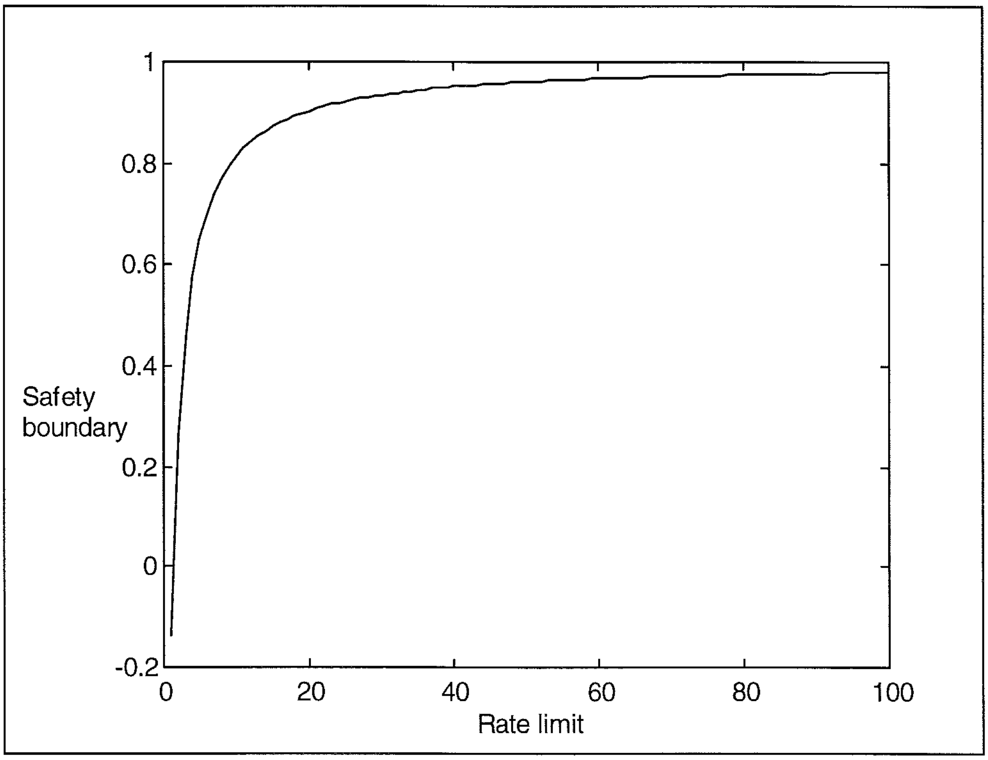


Figure 10
The safety boundary is drawn against the rate limit. For high rate limits the safety boundary tends to be the boundary that can be calculated without taking into account rate limiters

In case the rate limit is not too low, compared with the overall dynamic of the system, the method above described restricts minimally the freedom of the system and the response is similar to the one without rate limitations.
The above circuit can be used, being careful to pick from the table above an appropriate value for the safety threshold that the controller attempts to maintain.

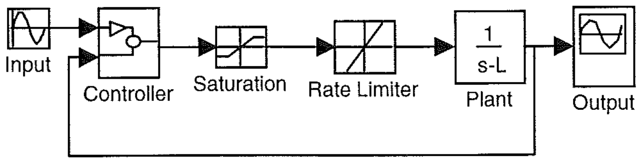


Figure 11
The block diagram is modified and a rate limiter is applied for simulation purposes
Modifications to the programmable controller are now necessary

For example, with a rate limit of 10, it is necessary to pick a safety threshold below 0.8, in the following figure a time response in this situation is showed :
Notice the small overshoots at the beginning of each plateau ; they are due to the presence of a rate limiter before the actuator.

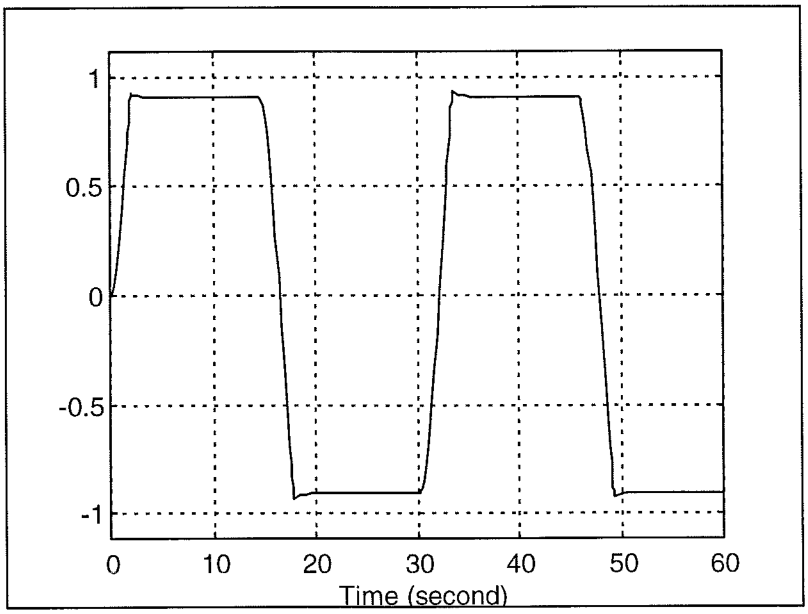


Figure 12
The simulation with rate limiters and increased boundaries in the controller
Notice the small overshoots due to the presence of the rate limiter

In case of a much tighter limit in the control rate and with very strong control inputs, still the system remains stable but a junky trajectory becomes much more visible as shown in the next figure.

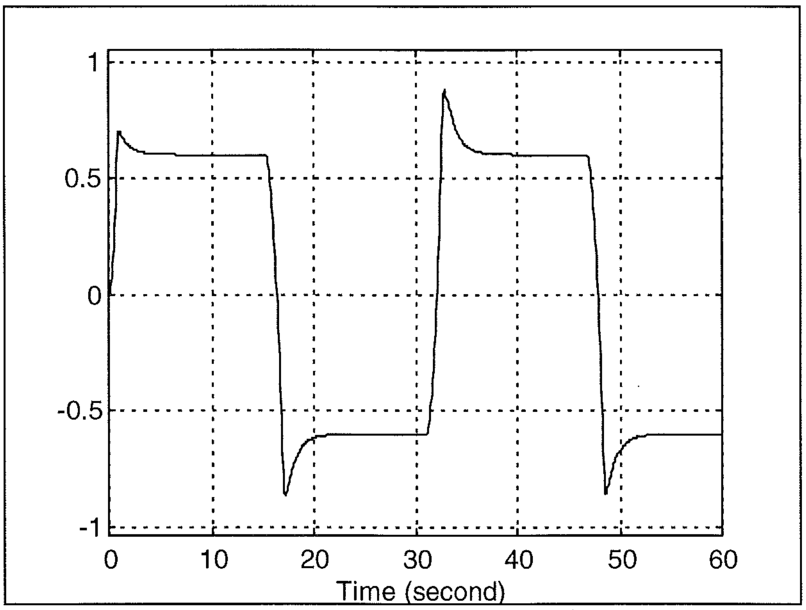


Figure 13
More severe rate limits make the overshoots to be more visible ; of course new boundaries need to be implemented to face these new characteristics of the system.

The long overshoots are the curves that were calculated before and represent the progressive reversing of the control input while the system keeps approaching the critical boundary.
Notice that the system is now working under severe restrictions that would not be necessary with a smarter logic.

4.4 A controller with dynamic boundaries

Lets take into consideration the equation of the previous paragraph, giving the last useful trajectory that the system can perform without falling into an uncontrollable status.

$$x = -\frac{\dot{u}}{\lambda^2} e^{\lambda T} + M + \frac{\dot{u}}{\lambda} T + \frac{\dot{u}}{\lambda^2}$$

Now let's substitute a generic time, that depends of the current status u of the input variable.

$$T = -\frac{u + \underline{u}}{\dot{u}}$$

which yields to :

$$x = -\frac{\dot{u}}{\lambda^2} e^{-\lambda\left(\frac{u+\underline{u}}{\dot{u}}\right)} - \frac{u}{\lambda} + \frac{\dot{u}}{\lambda^2}$$

At this point we obtained a relation that gives different boundaries according to the current state of the input.

If we plot the boundary versus the input status keeping the other parameters unitary and fixing a rate limit of 5 we obtain the following curve :

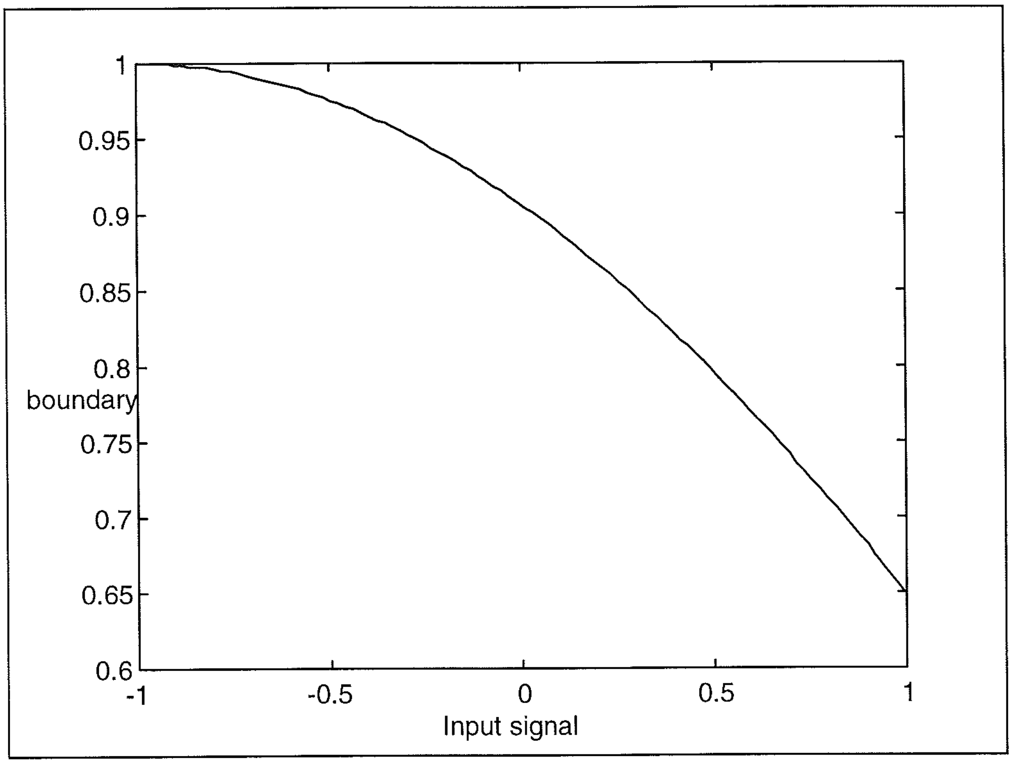


Figure 14
The last useful trajectory that keeps the system from trespassing the boundaries gives information on the maximum values allowed for the state variable for a given input condition.

We want now to implement this new relation in a controller that will use this relation to improve the performance of the system without endangering safety.
We now need a controller that gets as an additional information its own rate saturated output to decide dynamically the most appropriate limit. A possible modified block diagram can look as follows :

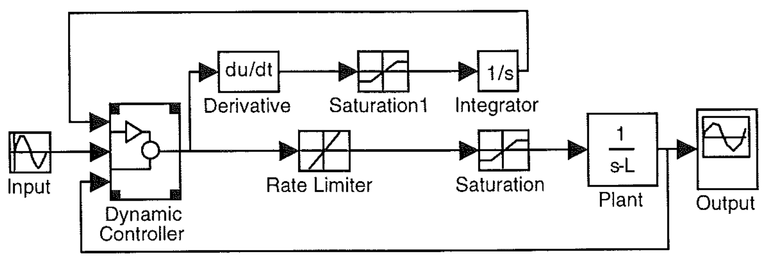


Figure 15
A way to gain information about the rate limited input

with the controller getting as an additional input its own rate-saturated output and using it to constantly compute the safety boundary after which no diverging control action is allowed.
Following is a simulation with a rate limit of 3 and, as usual an input saturation limited to one. Now the controller allows the system to fully use the whole [-1 +1] domain gently braking it when approaching the boundary limit even if the control input as fairly more than the double of the one allowed.

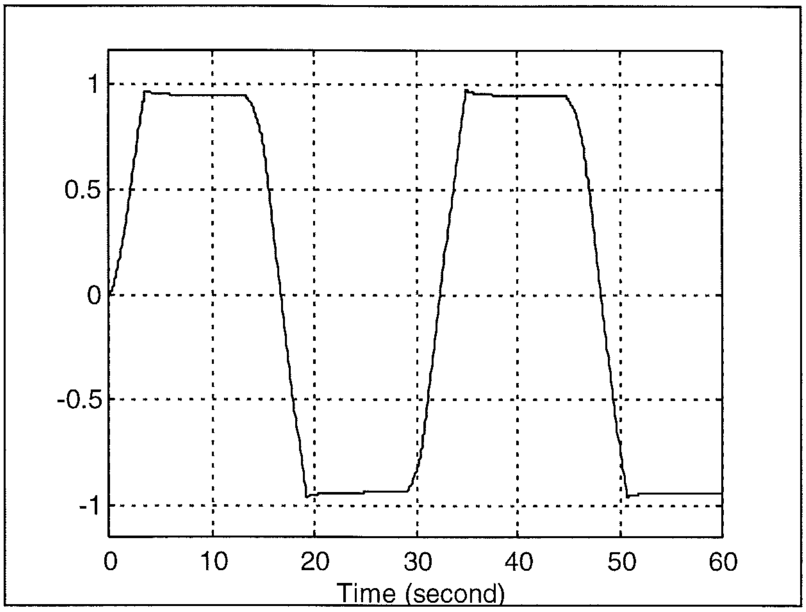


Figure 16
Simulation performed with the previous block diagram with the usual sinus wave as input.

The reason why the controller does not get the input directly after the rate saturator is because this would create an algebraic loop impossible to solve during the simulation.
The logic shown before seems to work but it is not really robust and tends to fail with very strong inputs or fast ones.
Another way to avoid the algebraic loop is to add a pole in the feedback and of course the bigger the pole is, the less it will affect the system's reaction bandwidth.
The new diagram will look as follows.

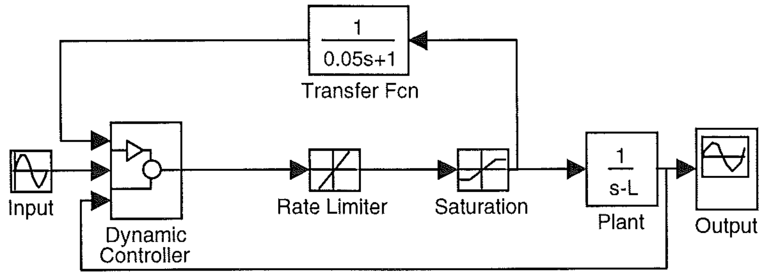


Figure 17
Taking the information after the rate limiter requires a device to break the algebraic loop ; in this case a low pass filter is used.

This new block diagram shows much more safety even with high inputs. However with fast inputs the controller fails to stop the evolution of the system in time, due to the limited bandwidth imposed by the feedback pole and of course putting the pole too far away multiplies the elaboration time enormously.

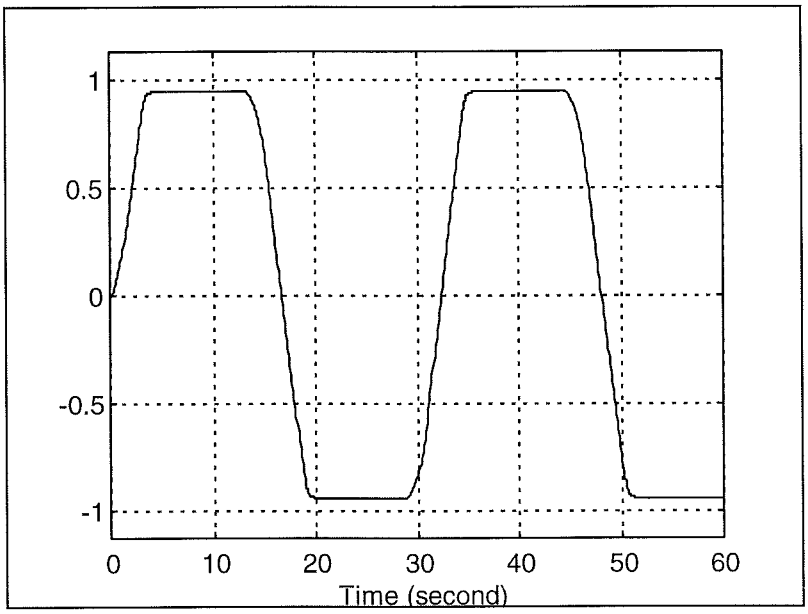


Figure 18
Taking the signal after the rate limiter results in a very accurate prediction of the trajectory.

Or a final way to efficiently break the algebraic loop is to insert a small delay in the feedback, so that rate saturated values are not straightway fed to the controller again, but are buffered in this example for one tenth of second, and then passed to the controller. The lack of a real-time feedback makes the algebraic loop no more existing. Notice that in the physical world is normal to have such small delays so having a minimal delay in the feedback adds even more realist to the simulation.

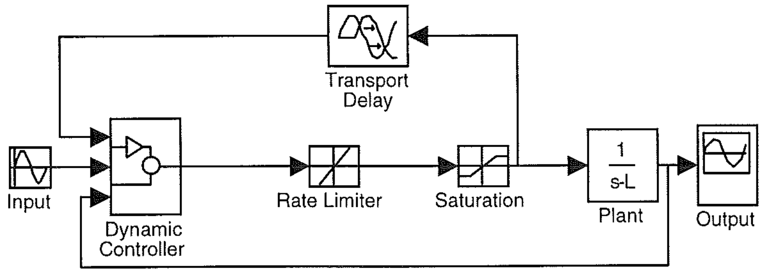


Figure 19
A minimal delay in the loop can also be used to break algebraic loops

The smaller the delay, the more precise will be the effect of the controller.
To simplify the controller's algorithm a linear approximation of the true limit curve was used. In the next picture the two boundaries are compared. Note that the linear approximation is anyway cautelative in respect to the exact curve.

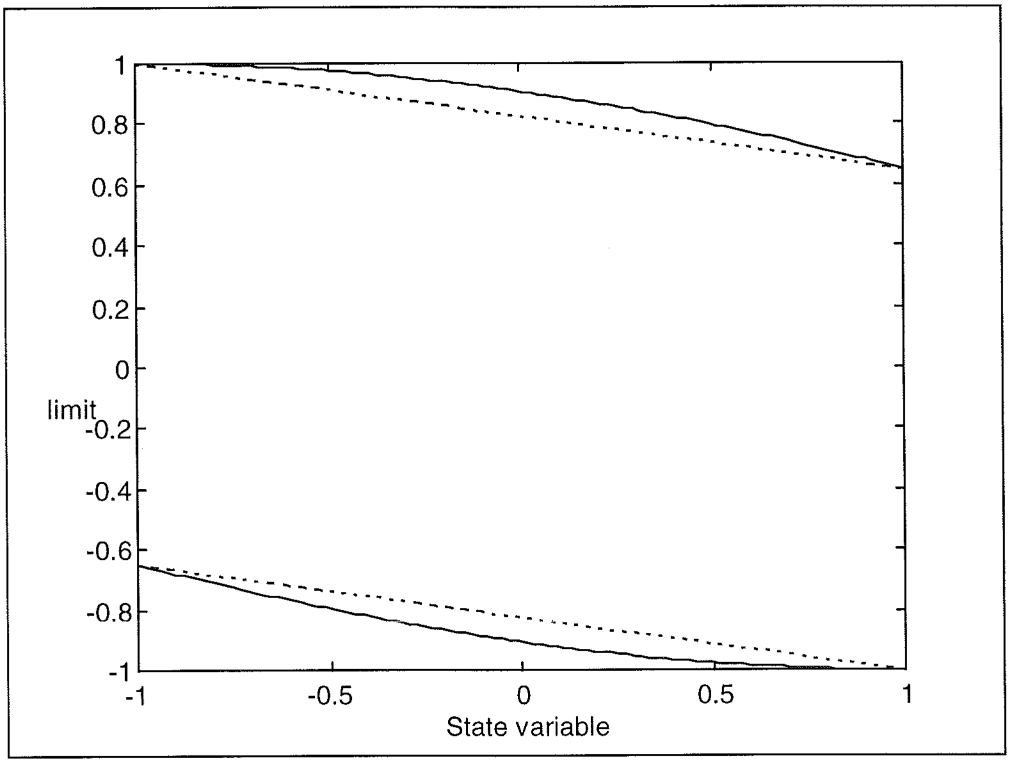


Figure 20
Safety boundaries may be approximated with segments, resulting in a cautelative error.

5. Pitch control of a JAS-39 model

5.1 Introduction

In this chapter we will focus our attention on multivariable systems characterized by unstable poles (i.e. with a positive real part) and affected by control saturations.
As already showed with single input/output system, also unstable MIMO system are characterized by a limited controllability envelope, whose shape and size is affected by the system's dynamic and by the actuator limits.
A linear model of the pitch dynamic of the JAS 39 Gripen will be used. The pitch dynamic of the JAS-39 is a very good example of a remarkably unstable system that needs control systems to be introduced as an interface between the pilot and the aircraft response. Several ways to estimate the operating limits will be discussed and several techniques to safely control the flight envelope will be tested on different degrees of complexity of the model, including saturation of the control input and afterwards introducing the additional issues that need to be considered when significant rate limiters affect the control loop.
Implementations of the techniques will be tested on a numerical example of the model and simulations results will be discussed.
Additionally the problem of the artificial stabilization will be discussed and several solution will be presented.
As a last issue, hybrid controlling will be implemented to mach safety boundaries and controllability envelope control with handling characteristics and flight quality requirements.

5.2 The flight model

We will consider a state-space model in the canonical form :

$$\dot{x} = Ax + Bu$$

where **x** is the state vector and contains the five state variables of the system in the form :

$$x = \begin{bmatrix} \alpha \\ q \\ \theta \\ \delta_e \\ \delta_c \end{bmatrix}$$

where α is the angle of attack, q the pitch rate, θ the pitch angle, δ_c the deflection of the canard wings and δ_e the deflection of the wing's elevators which are used in the dual role of ailerons and aft-elevators.

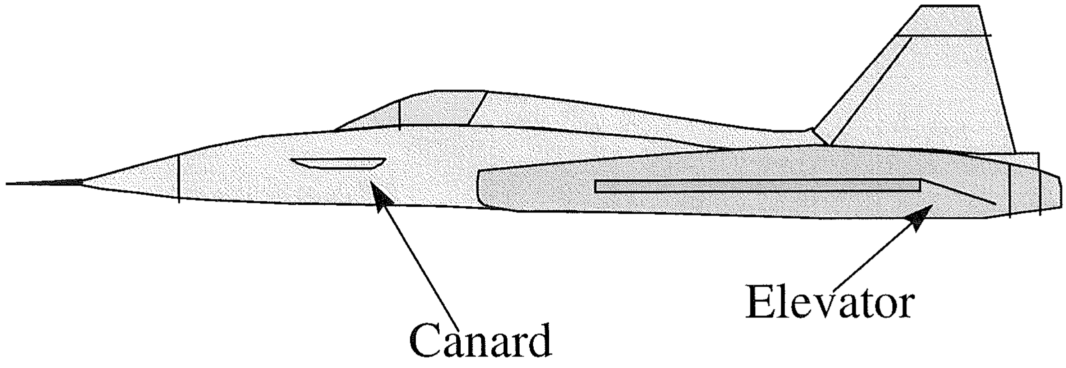


Figure 1

Control surfaces affecting the longitudinal dynamic on the SAAB JAS-39 "Gripen"

The vector **u** contains the system's external input and has the form:

$$u = \begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix}$$

where δ_{ec} and δ_{cc} are respectively the desired positions of elevators and canard wings, whose dynamic is represented by a first order system.

The matrix **A** is the characteristic matrix of the linear system and has the following structure :

$$A = \begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & * & 0 \\ 0 & 0 & 0 & 0 & * \end{bmatrix}$$

where the '*' mark non-zero matrix elements.

Obviously the third row is formed by zeros and only a unitary term since by definition :

$$q = \frac{d\theta}{dt}$$

while the 4^{th} and 5^{th} rows represent the first order dynamics of the control surfaces.

The vector **B** has the following structure :

$$B = \begin{bmatrix} * & * \\ * & * \\ 0 & 0 \\ * & 0 \\ 0 & * \end{bmatrix}$$

5.3 Stability analysis for the model

The matrix A will have, in general, 5 distinct eigenvalues either real, or complex conjugate. The real part of the eigenvalue will give information upon the stability of the associated mode ; if the real part is positive, the system will have an unstable mode, leading to a divergent time response in a non-perturbed motion with given initial conditions.
The particular model considered, will have five real eigenvalues, one of which remarkably unstable.
Let λ be an unstable eigenvalue, if v is the corresponding left eigenvector, by definition it is known that :

$$v A = \lambda v$$

now it is possible to change variables trough a linear transformation characterized by a matrix T such that :

$$z = T x$$

and the matrix T will be formed by five row-vectors in the following way :

$$T = \begin{bmatrix} v \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{bmatrix}$$

where v is the eigenvector considered and the other four vectors form together with v a group of linearly independent vectors.
Now, if we apply this transformation to the state space model given above, we obtain a new linear set of equations in a new set of state variables z that takes the form :

$$\dot{z} = T A T^{-1} z + T B u$$

which shows the relation between the new and old characteristic matrices.
Straight from the properties of eigenvalues and eigenvectors it is easy to see that the new system of equations will have the following characteristic terms that we will use in the following pages :

$$\dot{z} = \begin{bmatrix} \lambda & 0 & 0 & 0 & 0 \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix} z + \begin{bmatrix} \beta_1 & \beta_2 \\ * & * \\ * & * \\ * & * \\ * & * \end{bmatrix} u$$

where :

$$\begin{bmatrix}\beta_1 & \beta_2\end{bmatrix} = \nu B$$

doing so, and looking at the system just obtained, we can isolate the unstable mode of the system into a single scalar equation :

$$\dot{z}_1 = \lambda z_1 + \beta_1 \delta_{ec} + \beta_2 \delta_{cc}$$

in which z_1 is the only state variable.
Now and thereafter we will assume that in general the value and the derivative of the two control inputs are limited and such limitations will be defined as follows :

$$|\delta_{ec}| \leq \overline{\delta}_{ec}$$

$$|\delta_{cc}| \leq \overline{\delta}_{cc}$$

$$|\dot{\delta}_{ec}| \leq \Delta_{ec}$$

$$|\dot{\delta}_{cc}| \leq \Delta_{cc}$$

In addition, given the coefficients of the input signal, we can define the effective control signal δ for the unstable mode considered, which will be given by the expression :

$$\delta = \beta_1 \delta_{ec} + \beta_2 \delta_{cc}$$

and for this new control variable we can as well define the limits in terms of absolute value and derivatives, which will be function of the previously defined limits, according to the expressions :

$$\overline{\delta} = |\beta_1| \overline{\delta}_{ec} + |\beta_2| \overline{\delta}_{cc}$$

$$\overline{\Delta} = |\beta_1| \Delta_{ec} + |\beta_2| \Delta_{cc}$$

with which is possible to calculate the limits of the effective control signal δ.

5.4 Stabilizing the system with a state feedback

The pitch dynamic, due to the unstable pole, is difficult to control by a human operator. Usually a state feedback is applied in a inner control loop to shift the poles to more desirable locations, according to stability requirements, handling qualities and other considerations.
As already said the system considered has 5 real poles. Two of them ,very damped, represent the first order dynamic of the control surfaces, a third one, which has an almost

zero value, represent the almost indifferent behavior that the pitch dynamic shows towards different angles of pitch.
In a symbolic representation of the state matrix as a function of traditional stability derivatives of the aircraft, the theoretical value of this eigenvalue is exactly zero, since one row of the matrix would be entirely formed by zeros.
The final couple of eigenvalues, represents what is generally called 'short period' in a conventional airplane.
Generally, in the longitudinal dynamic of a conventional aircraft it is possible to point out two complex conjugate eigenvalues characterized by a high damping ratio as well as a high natural frequency. This mode, typical in every conventional-tail airplane is represented by a rapid pitch movement, strongly damped, with little speed change and is mainly caused by the aft tail (if any, of course).
The two real eigenvalues left in our model, one stable and one unstable, are the degeneration of such complex conjugated poles in a very unstable configuration. A similar effect can be obtained as well in a conventional airplane moving considerably the center of gravity.
In the next figure typical positions for such poles are shown together with the path of short period poles from a stable to an unstable configuration.

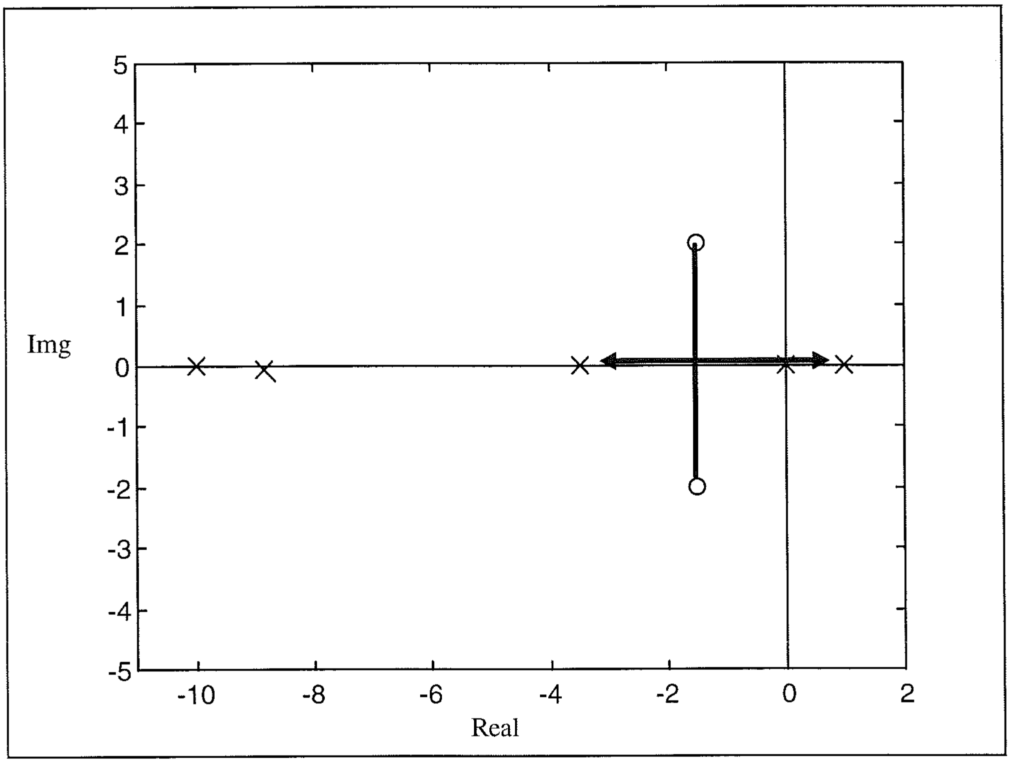


Figure 2
Typical position for the five eigenvalues are shown (X's). The two circles indicate expected positions for standard short period eigenvalues in a conventional airplane.

The purpose of an artificial stabilization system is to change the aircraft response, into a favorable one. The pilot's impression of the aircraft's ease of flight is determinant to define the flight quality of the airplane.
A pilot builds a subjective opinion of the aircraft's handling qualities in terms of how much the plane is easy to fly and how much the plane reacts satisfactorily to pilot's commands.

Several literary resources indicate guidelines to link this subjective average judgment with specifications on the poles of the linear model of the airplane.
According to the aircraft class and the flight condition, there are conditions for the damping ratio and the natural frequency of the short period that should be satisfied for the aircraft to be considered good in respect to handling qualities.
The more the real values are far away from these 'optimal' values, the lowest will be the flight quality of the aircraft.
The aircraft's dynamics are affected by structural properties, like the aerodynamic design, the weight distributions and so on.
During the early stage of the aircraft design, its possible to choose particular configurations with the aim of having a satisfactorily aircraft dynamic. Unfortunately most of the times stability and ease of flight of an aircraft must be traded with performance in terms of fuel consumption, maneuverability, speed and so on.
Usually general aviation airplanes or flight trainer, are designed to have aerodynamically stable profiles and good handling qualities, even if this penalizes performance.
On the other hand, an airplane such as the JAS-39 Gripen is designed for maximum performance and the designer traded all the flight qualities of the aircraft to maximize flight performance, mission range, speed and lifting efficiency.
The result is an aircraft that is almost impossible to be manually piloted, especially at low numbers of mach, when the center of lift is at it is most forward position and is closer to the center of gravity.
The highly damped short period mode, traditionally present in every conventional plane with tail horizontal stabilizers is turned into two real poles, one of which is unstable enough to seriously impair a human operator trying to manually stabilize the system.
Since the aerodynamics of the JAS-39 are designed to maximize performance, it is not possible to improve the handling qualities changing the flight design ; instead it is necessary to provide the plane with an artificial stabilizing system that, through the use of an internal control loop, changes the flight dynamic of the aircraft and present to the pilot's command an improved behavior of the system.
In our particular case, it is necessary to turn the two real poles into a pair of complex conjugate ones, with given damping ratio and natural frequency.
Several techniques may be used, all resulting in a feedback matrix that gets the state of the system and according to the value of the state variables gives an input signal contribution to the system.
In the following paragraphs several approaches will be tested to find optimal feedback matrices that result into desired behaviors without making too much use of the control action.
It is important to notice, that the input signals are physically limited and the pilot action shares the same signals with the artificial stabilizing system. If this last one make excessive use of input commands, may leave very little freedom to the pilot that will therefore feel very little authority on the aircraft evolution.

5.5 Controllability boundaries without rate limitations : Emergency recovery

As already noticed, the system is affected by an unstable mode whose analytical expression is :

$$\dot{z}_1 = \lambda z_1 + \beta_1 \delta_{ec} + \beta_2 \delta_{cc}$$

As already shown in the previous chapter, the presence of saturated control inputs define and limit the possible states that the system can assume without becoming uncontrollable. In this case z_1 has to fulfill the following condition :

$$|z_1| \leq \frac{\overline{\delta}}{\lambda} = z_{max}$$

that requires a controller that keeps the unstable mode inside the controllability envelope. Regardless of internal loops stabilizing the system, it is not possible generally to assure that the control system action mixed with the pilot's input will not drive the state beyond the controllability envelope.
Therefore it necessary to design a system that checks the value of the z_1 variable during the evolution of the system and behaves differently according to its value.
An example of such a logic may be found in the next table :

Value of z_1	Status	Action
$-z_{max}\,p < z_1 < z_{max}\,p$	Normal flight condition.	The internal feedback and the pilot's input are threaten normally. No other action is required.
$z_1 > z_{max}\,p$	Auto recovery : Emergency push -down	z_1 is positive and its module is dangerously high. The control loop and the pilot inputs are broken ; the control surfaces apply full negative control action until the system exits from this condition.
$z_1 < -z_{max}\,p$	Auto recovery : Emergency pull-up	z_1 is negative and its module is dangerously high. The control loop and the pilot inputs are broken ; the control surfaces apply full positive control action until the system exits from this condition.

where 'p' is a safety parameter, positive and less than one in value, that states how much the boundary proximity is overestimated in respect to the theoretical one.
If correctly tuned this system may avoid the pilot to drive the plane into uncontrollable situations, simply cutting off pilot's authority when limits are close to be exceeded. This of course adds safety to the flight maneuvers but at the same time it is not really desirable a control system that leaves everything in the hands of the pilot waiting for the very last moment to grab the control of the whole plane overriding the pilot's input in a moment in which maybe the pilot command is critically important (for example, low level flight, final landing phase, weapon aiming and delivery and so on).
In the next paragraphs more flexible control systems will be tested in the attempt to smoothen the transition between full-pilot-command and full-recovery-command.

5.6 Creating a short period

As already explained, the aircraft model bears two real poles, one highly stable and one highly unstable, that substitute the conventional short-period.
We will now try to move the two poles of the short period into desired positions trying not to affect the other system's modes.

In the next paragraphs will prove useful to redefine the matrix T as a square matrix whose rows are the five eigenvectors of the matrix A.

$$T = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix}$$

Such a matrix allow us to transform the system into a complete diagonal one, where all the variables of the vector z represent the five first order modes.
Changing variables the new system will look like :

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \\ \dot{z}_5 \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 & 0 \\ 0 & 0 & 0 & \lambda_4 & 0 \\ 0 & 0 & 0 & 0 & \lambda_5 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{bmatrix} + \begin{bmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \\ \beta_{31} & \beta_{32} \\ \beta_{41} & \beta_{42} \\ \beta_{51} & \beta_{52} \end{bmatrix} \begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix}$$

And the interaction between the various system modes is only due to the sharing of the same control actions. Without external control inputs the five state variables will evolve independently according to their respective initial conditions.
We will now focus our attention on the two poles that represent a conventional aircraft's short period. Namely we will assume that they are the first two ones, λ_1 being the unstable one and λ_2 being the stable one.
The system we will consider will therefore be a simple second-order one :

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \end{bmatrix} \begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix}$$

The matrices and vectors of the above system will be defined with new symbols and the system will be written as :

$$\dot{Z} = \Lambda Z + \Phi u$$

where Z can be obtained from x through the transformation :

$$Z = T_R\, x$$

where T_R is a matrix formed from a subset of the T matrix :

$$T_R = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

Now we may think of changing the position of the poles with a simple pole placing. What we will obtain is a feedback matrix that will change our two poles into a conventional short period with complex conjugate poles.
However, since the system has two input, the pole placement problem will have ∞ solutions. Therefore we have one degree of freedom to play with when deciding the inputs generated by the feedback matrix.
What we will do now will be to decide *a priori* a ratio between the elevator input and the canard input signals to reduce the pole placement problem to a single solution problem.
To be sure that our stabilizing system will affect as much as possible the unstable mode and at the same time will not affect too much the other modes, the control inputs will be proportional to their respective β coefficient that quantifies their effect on the unstable mode.

$$\begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix} = \begin{bmatrix} 1 \\ \dfrac{\beta_{12}}{\beta_{11}} \end{bmatrix} \delta_{ec} = \alpha\, \delta_{ec}$$

and the second-order system will become :

$$\dot{Z} = \Lambda Z + \Phi \alpha \delta_{ec}$$

that is a single input system.
Now, imposing a pole placement to this system will result into a unique k feedback vector that will in our objectives gently move the two poles without turn the whole system's modes upside down.
The stabilized second order system will have the analytical expression :

$$\dot{Z} = \Lambda Z - \Phi \alpha k Z = (\Lambda - \Phi \alpha k) Z$$

where the matrix inside the parenthesis will have the desired eigenvalues.
In the original fifth-order state space system, we can use a feedback matrix K that is linked to the feedback vector k by the simple relation :

$$K = \alpha k T_R$$

and will change the original system into :

$$\dot{x} = A x - B K x + B u' = (A - B K) x + B u'$$

where u' is the input by the external operator (the pilot or the navigation system).

5.7 Single pole placement

In this paragraph we will develop a technique to modify as sharply as possible only the unstable mode of the system, leaving the others almost totally unaffected.
Using the same notation of the previous paragraph, the scalar equation of the unstable mode is expressed by :

$$z_1 = \lambda z_1 + \beta_{11} \delta_{ec} + \beta_{12} \delta_{cc}$$

Again we want the feedback action to focus the action on the unstable mode and therefore we will link again the two control inputs by a proportion according to their effectiveness on the mode.
At the same time we want to stabilize the system moving the position of the pole on the other side of the imaginary axis.
By defining the feedback as :

$$\begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix} = -p_1 \begin{bmatrix} \beta_{11} \\ \beta_{12} \end{bmatrix} z_1$$

we can arbitrary move the real positive pole all over the real axis, simply changing the weight p_1.
The new system will have the expression :

$$\dot{z}_1 = \left[\lambda_1 - p_1(\beta_{11}^2 + \beta_{12}^2)\right] z_1 + \beta_{11}\delta'_{ec} + \beta_{12}\delta'_{cc}$$

were again the δ' terms are external inputs.
It is worthwhile to notice that is very easy to set up the new value for the pole. The system will be stable if :

$$p_1 > \frac{\lambda_1}{\beta_{11}^2 + \beta_{12}^2}$$

It is easy to see that the overall feedback matrix for the original system can be expressed as :

$$K = (T\,B)^T\,P\,T$$

where P has the structure :

$$P = \begin{bmatrix} p_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The final system will have all real poles and the position of the other four ones will not be touched by the feedback system.
Given an input value to the system, the unstable mode will behave as a stable first order system and will approach regularly an asymptotically a steady-state value without overshoots.
Therefore this placement is very much suited when the value of the unstable mode has to be carefully controlled. On the other hand, the handling characteristics of this system are extremely poor, since the short period is formed by two stable real poles.
One last comment deserves the attempt to use this method to change two eigenvalues at the same time.

Given the structure of the matrix P, we may think of inserting two weights to change the unstable mode together with the marginally unstable one in the following way :

$$P = \begin{bmatrix} p_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & p_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

the idea was to have p_3 as a very small weight, just enough for the marginally unstable mode to become marginally stable. Unfortunately the two inputs allow only one degree of freedom (there is only a single redundancy), and therefore only one pole can be changed without affecting the others. Moving two poles in this way result into a strong interaction that can lead to unsatisfactory results if the weight p_3 is anything more than very little.

5.8 A numerical example of the model

In this paragraph we will present a numerical example of the model given above. With this numerical example we will try to synthesize several kind of controllers to achieve good handling characteristics together with closed loop stability, using the techniques described above.
The matrix A, of the state space model will have the following value :

$$A = \begin{bmatrix} -1.3936 & 0.9744 & -0.0019 & -0.5349 & -0.0071 \\ 5.687 & -1.1827 & 0.0002 & -25.9398 & 7.9642 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix}$$

while the control input will be characterized by the following matrix B :

$$B = \begin{bmatrix} 0.0676 & -0.0313 \\ 1.527 & 0.4002 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

and both the matrix reflect the typology described in the second paragraph.
The matrix has five real eigenvalues :

$$-20 \quad -20 \quad 0.0027 \quad 1.0662 \quad -3.6452$$

three of which are stable, and two are unstable.
The five eigenvalues are written in the rows of the following matrix T :

$$T=\begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ -0.7771 & -0.1908 & 0.5310 & 0.2682 & -0.0757 \\ -0.8098 & -0.3502 & 0.0014 & 0.4518 & -0.1321 \\ -0.8036 & 0.3182 & -0.0004 & -0.4784 & 0.1553 \end{bmatrix}$$

The two highly stable eigenvalues represent the dynamic of the two control surfaces, while the third one is almost zero and represent the almost indifferent equilibrium condition in respect to the pitch angle.
The last two eigenvalues represent the short period and one of them is remarkably unstable.
If we now apply the transformation changing the five variables into the five directions of the modes of the system, we obtain the diagonal system :

$$\dot{z}=TAT^{-1}z+TBu$$

which numerically has the following form :

$$\dot{z}=\begin{bmatrix} -20 & 0 & 0 & 0 & 0 \\ 0 & -20 & 0 & 0 & 0 \\ 0 & 0 & 0.0027 & 0 & 0 \\ 0 & 0 & 0 & 1.0662 & 0 \\ 0 & 0 & 0 & 0 & -3.6452 \end{bmatrix} z+\begin{bmatrix} 20 & 0 \\ 0 & 20 \\ 5.0205 & -1.5659 \\ 8.4471 & -2.7576 \\ -9.1357 & 3.2582 \end{bmatrix}\begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix}$$

with the five eigenvalues forming the diagonal of the system matrix.
Now we can implement one of the techniques to obtain a feedback matrix K that stabilizes the system.
Analytically the matrix K will turn the state space system from :

to :

and the new matrix A-BK will have to satisfy our requirements.
As already introduced, there are in literature a number of different specifications of the poles of the aircraft that define the handling qualities in terms of the pilot impression of the aircraft's response and flying ease.
In our case we will try to apply some of these considerations using the diagram of the next figure.
On the two axes are reported the natural frequency and the damping ratio of the short period and the area is divided into zones of different qualities.
It is obvious that our aim will be to bring the eigenvalues corresponding to the short period into the inner sector, so that we will give good handling qualities to the plane, when closed in the feedback loop.
If use a pole placement technique, we are sure to move the poles wherever we want and therefore we can decide arbitrarily the handling qualities of the short period.
Unfortunately the resulting K matrix will usually make excessive use of the control inputs.
The problem is that any linear optimization does not take into account the presence of

saturations and in some cases even when the system is close to the equilibrium point the controller may order an excessive input.

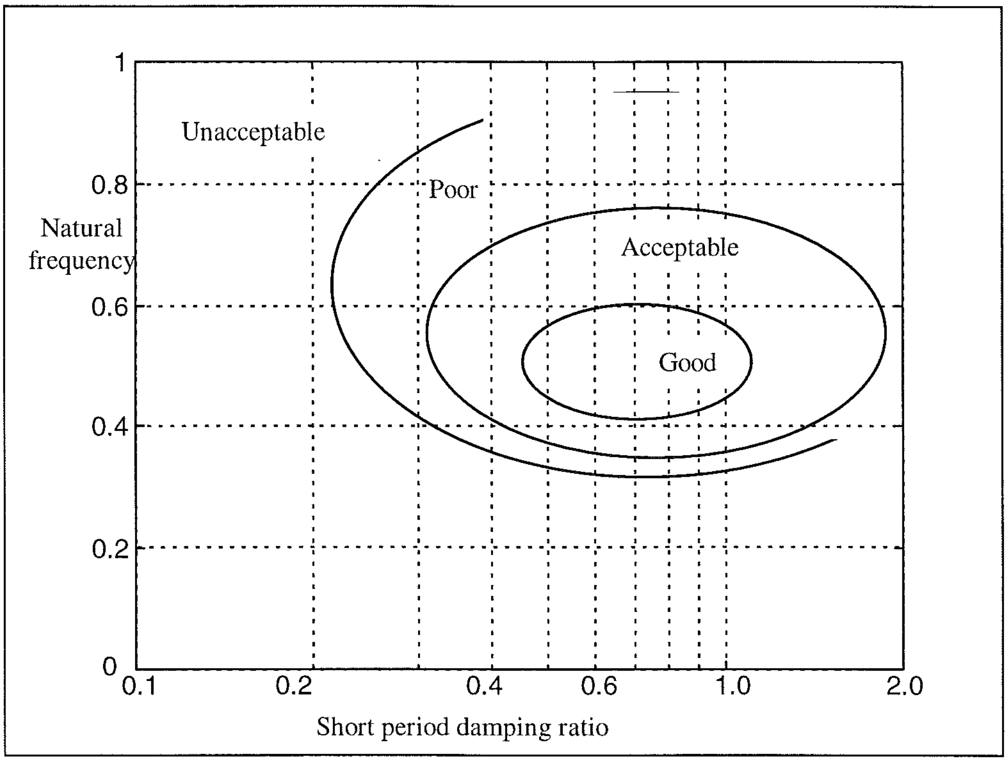


Figure 3
Typical relations between the handling qualities and the short period

On the other hand, an optimization like the LQR/LQG does not give control over the poles of the system. The result of this is sometimes an undesired shift of the other poles, and a change of the eigenvectors into undesirable positions.
We will try instead to use the method described in the previous paragraphs to create a stable short period with desirable characteristics.
First of all we will aim to a couple of complex conjugate poles with the value :

$$-1.27 \pm 2.52$$

which corresponds to point A on the next figure.
The corresponding handling quality is therefore good and the little damping ratio is well suited for a fighter aircraft.
To obtain these poles we will proceed as explained before. First of all we have to extract some values to create the small matrices of the second order system :

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} \lambda_4 & 0 \\ 0 & \lambda_5 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \end{bmatrix} \begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix}$$

that can be written also as :

$$\dot{Z} = \Lambda Z + \Phi u$$

Looking at the numerical values it is immediate to extract the following values :

$$\Lambda = \begin{bmatrix} 1.0662 & 0 \\ 0 & -3.6452 \end{bmatrix}$$

$$\Phi = \begin{bmatrix} 8.4471 & -2.7576 \\ -9.1357 & 3.2582 \end{bmatrix}$$

and immediately we can calculate the proportion between the two inputs that will be used by the control system :

$$\begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix} = \begin{bmatrix} 1 \\ \dfrac{\beta_{42}}{\beta_{41}} \end{bmatrix} \delta_{ec} = \alpha\,\delta_{ec} = \begin{bmatrix} 1 \\ -0.3265 \end{bmatrix} \delta_{ec}$$

and it is now a straightforward to calculate with a pole placement the final K_A matrix that will be :

$$K_A = \begin{bmatrix} -0.3399 & -00273 & 0.0003 & 0.0230 & -0.0040 \\ 0.1110 & 0.0089 & -0.0001 & -0.0075 & 0.0013 \end{bmatrix}$$

which can implement this matrix into a feedback controller.

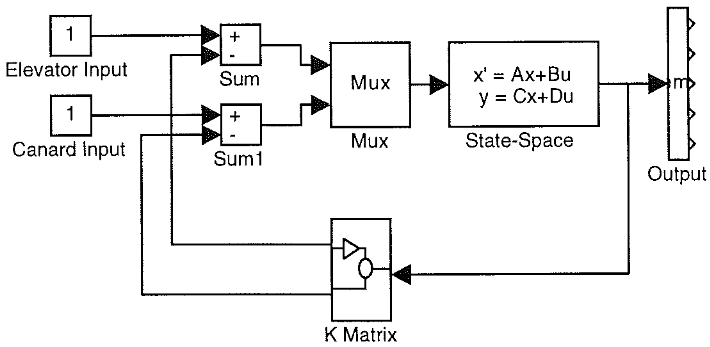


Figure 4
State space model with matrix feedback

In the previous picture a linear model is implemented for simulations.
Since the model is linear, the response will be the expected one regardless of input amplitude and history.
A more interesting case is when input saturations are applied to the control inputs, reflecting that the control surfaces have a limited excursion.
We will assume for the future a limit of 30° both for ailerons and canard surfaces ; notice that the above limit can not simply be overcome giving more movement freedom to these surfaces since greater angles will not produce larger increases in the effect due to separation of flows around the surfaces.

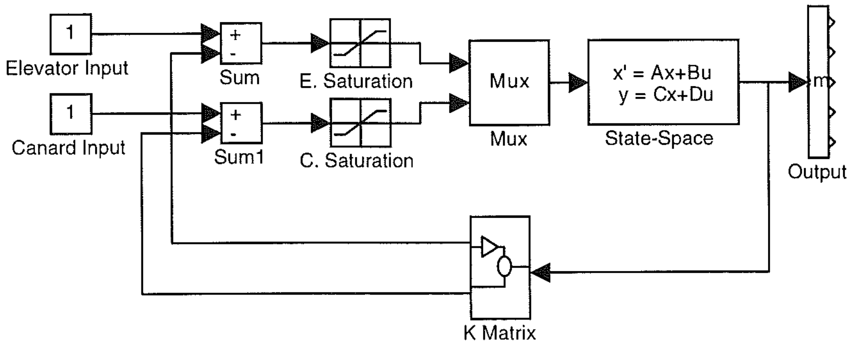


Figure 5
Input saturation are added in the stream of the Canard and Elevator signals

With this new scheme, we run the risk of driving the system towards the controllability edges.
As already showed in the previous paragraphs it is possible to extract the value of the unstable mode z_4 as a linear combination of the five state variables.
In this case :

$$z_4 = \begin{bmatrix} -0.8098 & -05032 & 0.0014 & 0.4518 & -0.1321 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

To calculate the stability margins we have to look for the diagonal system.
In the same way it was shown for first order systems, if z_4 exceed certain values it is impossible with the limited input available to gain control of the system which will start to diverge indefinitely.
These boundaries are such that:

$$|z_4| \leq \frac{\overline{\delta}}{\lambda} = z_{\max}$$

which numerically means :

$$|z_4| \leq\leq 315.2763$$

Another kind of feedback matrix can be found just moving one pole, as showed at the beginning of this chapter.
We have one highly unstable eigenvalue, and we will now move it, while still keeping it on the real axis, so that it will assume a negative value.
By defining the feedback matrix K such that :

$$K = (T\,B)^T\;P\,T$$

where P is a matrix with only one non-zero term :

$$P = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & p_4 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

such feedback matrix will change the unstable mode into the following new scalar equation :

$$\dot{z}_4 = \left[\lambda_4 - p_4(\beta_{41}^2 + \beta_{42}^2)\right]z_4 + \beta_{41}\delta'_{ec} + \beta_{42}\delta'_{cc}$$

were the δ' terms are external inputs, and we can stabilize it simply choosing any value for p_4 such that :

$$p_1 > \frac{\lambda_1}{\beta_{11}^2 + \beta_{12}^2}$$

choosing p=0.026 will result for example into an eigenvalue of :

$$\lambda_4 = 0.9867$$

The corresponding K_C matrix will be :

$$K_C = \begin{bmatrix} -0.1778 & -0.0769 & 0.0003 & 0.0992 & -0.029 \\ 0.0581 & 0.0251 & -0.0001 & -0.0324 & 0.0095 \end{bmatrix}$$

and bears several desirable properties.
This matrix in fact turns the unstable mode into a stable first order one and applying the maximum input allowed will result in the unstable mode to asymptotically tend to a value very close to the allowed limit without overshoots.
This means that no matter of the input the linear model will always remain inside the controllability envelope.
However, even if the linear model never exceed the limits, if we introduce saturations in the control loop then surprisingly the system will not be able to keep between the borders.
The reason for this is that the control loop drives the system to tend asymptotically to a value that is smaller than the allowed limit but drives it using control signals that saturate one of the two actuators.
It is necessary then to have a controller that compensate with the other actuator if one of the two actuators is saturated and this can be done with a cross-feed circuit modification showed in the next figure.
Basically the cross-feed transfer the exceeding input to the other input after having it multiplied by a coefficient which is different according to the direction to which the input is transferred.
A problem raises since the transfer functions of the canard and the elevator are somewhat different ; it is not possible to make them equal since the two transfer functions are both

non minimum phase systems and it is not possible therefore to turn one of them into the other.
On the other hand these differences are not too strong and in a way it is possible trade one of them for the other if we don't expect an exactly equal behavior.

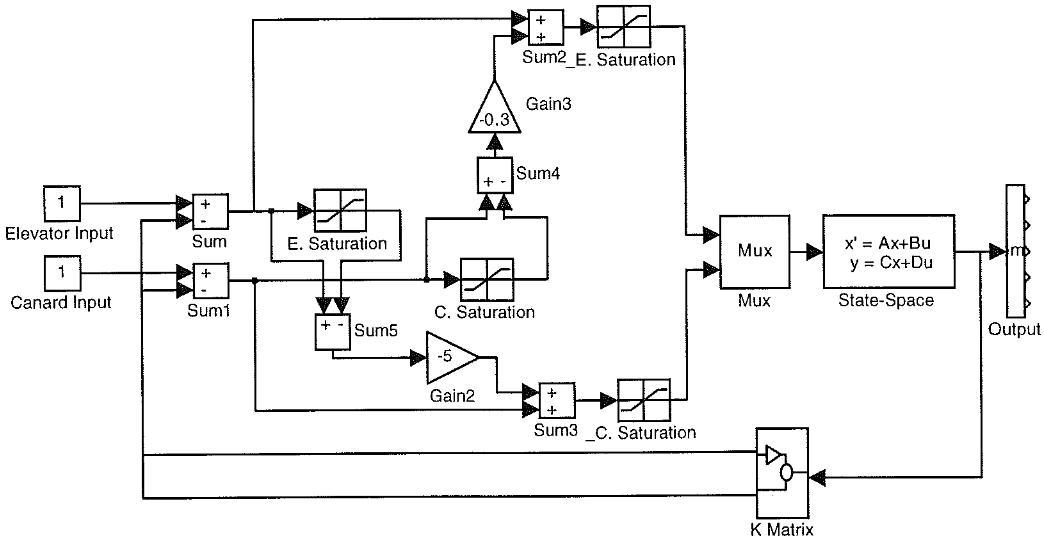


Figure 6
Cross-feed of saturated inputs to compensate for single surface excursion saturation

In the case of this circuit, the problem is solved transferring an excessive amount of signal. If we look at the modules of the transfer functions, we see that the elevator is more than four times more effective than the canard winglet comparing them with an equal angle of attack. The reason for this is that the canard winglets are not really too much involved in the control of the plane, but instead their biggest purpose is to act as vortex generators allowing the wing behind to sustain a much higher angle of attack before suffering an aerodynamic stall[1].
In the model redundancy of the signal is obtained therefore multiplying by 5 any signal exceeding from the elevator and directed to the canard while dividing by 3 any signal that saturates the canard movement and it is therefore redirected to the elevator actuator.
In this way, when one of the control surfaces will be saturated, the overall action of the controller will instead be even stronger, thus compensating for any asymmetry due to the fact that only one control surface is used.
As a final step, to make the model more flexible , we will substitute any part of the controller with a programmable box that takes the two external inputs and the five state variable and returns directly the input signal directed to the two actuators.
In this way, as done in the previous chapter, we will turn our block diagram into a very simple one, shown in the next figure ; any kind of control techniques or any special modification to the block diagram will be implemented changing the software of the control box.

[1] The vortex produced by the canard winglet will give more kinetic energy to the lower layers of air that are in direct contact with the wing surface. It is the low energy of these layers that are usually associated with the end of the laminar flow around a surface. The vortex basically mixes layers in a vertical motion that distributes more evenly the energy.

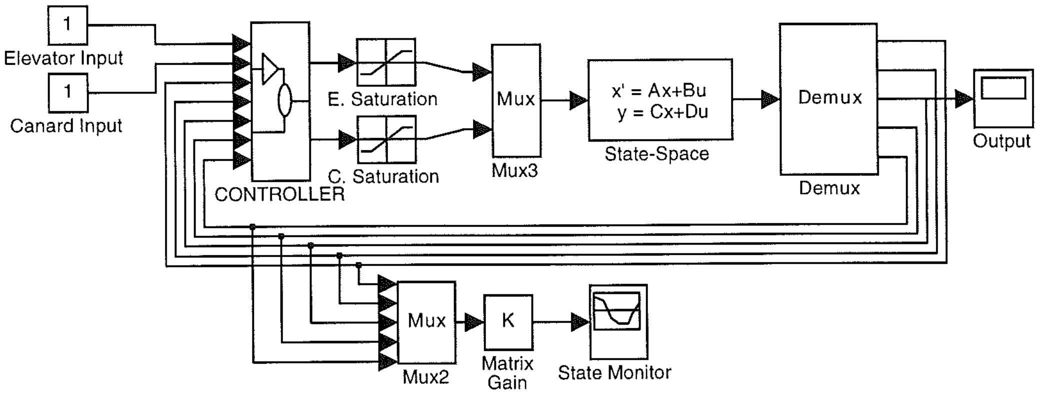


Figure 7
The state space model with a programmable controller and a state monitor that extract information on the state of the unstable mode.

Now the model is ready for various simulations to be performed, with different feedback matrices and control strategies.
In the control box can also be included a recovery logic that forces the maximum converging input to the system in case the state variable of the unstable mode reaches certain limits, as already done in the previous chapter.

5.9 Hybrid control

In the previous paragraph we used two different approaches to get two different state feedback matrices. To obtain the first one, we looked to handling requirements and we wanted the short period poles to be locates in certain well-defined positions.
This feedback matrix called K_A does not take into account the presence of saturation in the control inputs and therefore it may not bear desirable behavior nearby the saturation limit.
The other matrix, K_C, was instead designed keeping only the saturation limit as a reference point and without disturbances keeps always the system within boundaries.
The problem with this matrix is that it keeps the eigevalues of the short period on the real axis and therefore the resulting handling qualities are among the worst ones.
It is clear that these two matrices are desired in different flight condition and it could be useful to have K_A acting around neutral positions and far away from dangerous states, giving the optimal handling feeling to the pilot and having K_C as a feedback matrix during critical assets to be sure that while still controllable by pilot's inputs the system will keep inside the given boundaries.
Of course it is not reasonable to have an abrupt passage from one way to another way of controlling, but instead the two matrices should mix gradually, linearly combined with a parameter that depends on the value of the unstable mode z.
Of course membership functions of the fuzzy controlling comes very useful in the designing of suck a controller ; in a given point we can have that the instantaneous feedback matrix that is expressed by the expression :

$$K = q\,K_A + (1-q)K_C$$

where q is a function of the unstable mode state variable.
In general the full domain of the z variable can be divided into area of influence of the various matrices in a fashion depicted by the next figure.

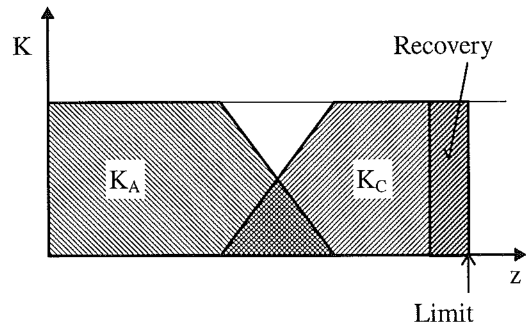


Figure 8
Fuzzy logic may help the controller in decide which feedback matrix should be used and how to combine them to obtain a dithering effect from one to another according to the value of one (or more) state variable.

Which shows according to the state which are the weights of the matrices.
The last block represents the extreme recovery action done by the recovery logic already implemented several times before.
This last recovery logic breaks the control loop and forces converging actions ignoring all the other inputs. Notice that without any errors K_C itself should make this last recovery block unnecessary.
Before showing different ways to combine matrices, we will first of all calculate a third feedback matrix meant to be a sort of "half-way" between these two.
Applying the same method showed before for a double pole placement we can calculate a feedback matrix such that the short period eigenvalues are :

$$-1.5 \pm 2.01i$$

obtaining the corresponding feedback matrix K_B :

$$K_B = \begin{bmatrix} -0.3399 & -0.0273 & 0.0003 & 0.023 & -0.004 \\ 0.111 & 0.0089 & -0.0001 & -0.0075 & 0.0013 \end{bmatrix}$$

The three matrices just obtained present handling qualities that gradually decrease, starting with K_A which bears optimal handling qualities and ending with K_C with bears extremely poor handling qualities.
These three matrices can be combined in endless ways, always taking into account that K_A should be valid around the neutral positions and K_C should be valid nearby the operating boundaries.
The matrix K_B can be used optionally to have more control on the intermediate transition between the first and the last one.
The problem when doing this kind of mixes is that there is no analytical proof that the constantly time-changing linear combinations of these matrices will give expected results ; for example it leaves a feeling of greater reliability the use of plateaus in the design of the membership functions. A plateau leaves a feedback matrix fully active over a given range without any influence from other matrices ; this will ensure that at least in that range the system's behavior will be the one predicted, and it seems opportune to have this feature at least in the more critical states of the system.

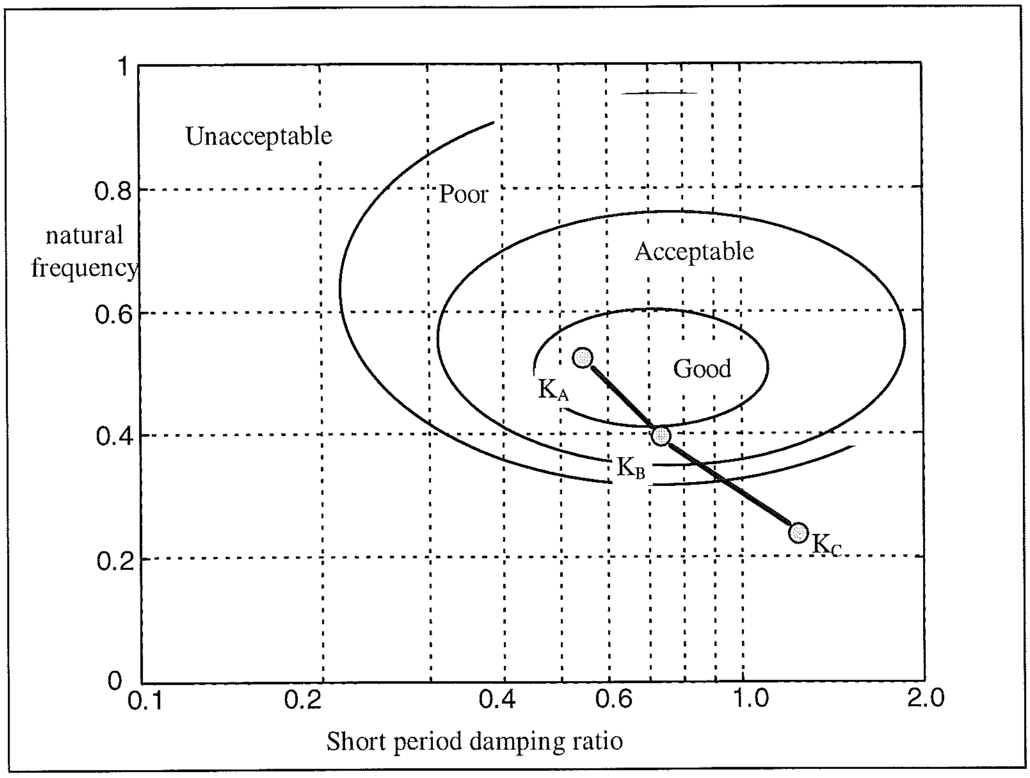


Figure 9
The three feedback matrices with their respective positions in the short period handling qualities diagram

what follows are several kind of mixed controllers that were tested in the simulations.

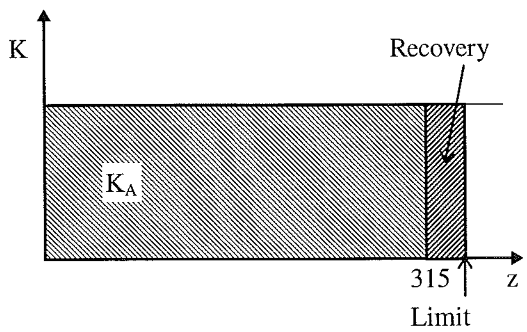


Figure 10
Case one : basic non-hybrid controller with only one kind of feedback matrix and an all-or-nothing recovery system at the end of the safety zone.
The system will be left free to evolve until the limit is reached ; once the limit is trespassed a recovery action will bring the system inside the safety zone again.

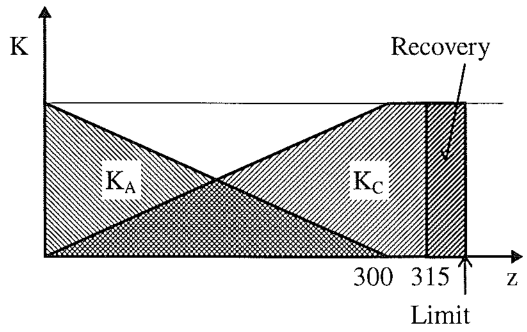


Figure 11
Case two : simple hybrid controller, the matrix K_A turns gradually into K_C, which is designed to make it impossible for the external operator to drive the system towards certain values of the unstable mode state variable. Notice that in the final part, before the recovery system zone, the feedback matrix K_C remains alone to ensure its safety effect on the control of the unstable mode.

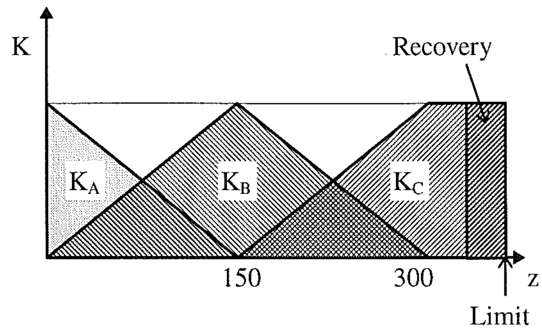


Figure 12
Case three : same as case two, but this time the matrix continuously passes through all the three feedback matrices considered. K_B allows more control on the intermediate steps between K_A and K_C.

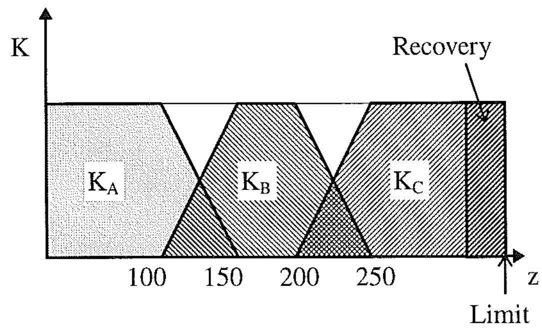


Figure 13
Case four : plateaus are inserted so that each matrix has a range within which does not suffer any kind of influence from the other ones.

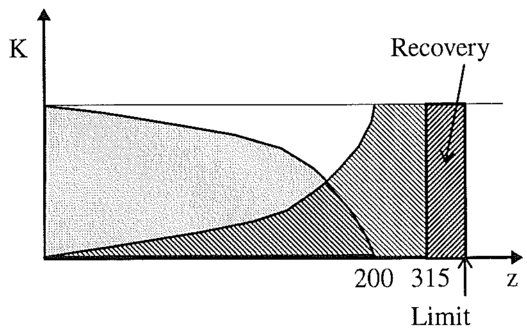


Figure 14
Case five : a parabolic pattern is used to mix the two matrices in a non-linear fashion.

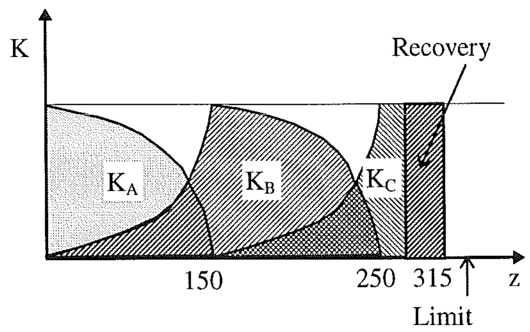


Figure 15
Case six : multiple parabolic patterns mix all the three matrices in a continuos fashion.

Obviously these are just some of the endless ways can be thought of when it comes to mix matrices. These examples where implemented in form of Matlab code and tested in real time simulation with manual inputs.

The overall results were positive, and the best ones proved to be the ones with a continuos change in the feedback matrices without any kind of intermediate plateau.
Particularly the parabolic shape gives the possibility to the main K_A matrix to remain almost untouched for a considerable range of the state variable z and then gradually but at an increasing speed it is replaced by the more safe K_C matrix, without any discontinuity.
Controllers with intermediate plateaus, instead, such as the one described in case four, showed in many cases an irregular behavior due to the fact that after a continuous change in the control matrix, it gets suddenly stuck into a value for a certain range and this change is much more visible than the continuos change and is also visible in case of simple step responses.

5.10 Effect of rate limiters

As already done in the previous chapter, we will now examine the operating envelope of the system, according to the existence of the unstable pole and given limits to the control signal's value and changing rate.
Given the equation :

$$\dot{z} = \lambda z + \delta$$

and given the limits :

$$|\delta| \leq \bar{\delta}$$

$$|\dot{\delta}| \leq \Delta$$

a new set of non-dimensional variables will be defined such as :

$$x_1 = \frac{\lambda z_1}{\bar{\delta}}$$

$$x_2 = \frac{\delta}{\bar{\delta}}$$

$$u = \frac{\dot{\delta}}{\Delta}$$

$$\tau = \lambda t$$

$$\gamma = \frac{\lambda \bar{\delta}}{\Delta}$$

the problem can now be expressed in terms of the new non-dimensional variables and will take the form :

$$\frac{d}{d\tau}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{\gamma} \end{pmatrix} u$$

with the already known limits, expressed as well in the new form :

$$|x_2| \leq 1$$

$$|u| \leq 1$$

Now it is interesting to gain information on the evolution of the system when the maximum control action is applied to the aerodynamic surfaces.
For u=1 the previous equation is satisfied by the family of curves given by the equation :

$$(x_1 + x_2 + \frac{1}{\gamma})e^{-\gamma x_2} = C$$

where C is an arbitrary constant.
In the same way for u=-1 we obtain a corresponding curve :

$$(x_1 + x_2 - \frac{1}{\gamma})e^{\gamma x_2} = C$$

Now it is interesting to consider for example the curves that intersect the origin ; their expression is :

$$x_1 = -x_2 + \frac{e^{\gamma x_2} - 1}{\gamma} \qquad \text{(u=1)}$$

$$x_1 = -x_2 + \frac{1 - e^{-\gamma x_2}}{\gamma} \qquad \text{(u=-1)}$$

while another interesting pair of curves are the ones that touch the controllability edge of the state variable x_1 without crossing it.
The upper one is the curve with u=-1 which crosses the point x_1=1 and x_2=-1 which has the expression :

$$x_1 = -x_2 + \frac{1}{\gamma}(1 - e^{-\gamma(x_2+1)})$$

while the lower one is the curve with u=1 which crosses the point x_1=-1 and x_2=1 and bears the following expression :

$$x_1 = -x_2 + \frac{1}{\gamma}(e^{\gamma(x_2-1)} - 1)$$

these relations can be used to compute a restricted range for z_1. For example if we substitute the value x_2=1 in the first boundary equation we obtain the value for x_1 :

$$x_{1MAX} = -1 + \frac{1}{\gamma}(1 - e^{-2\gamma})$$

and in the same way substituting x_2= -1 in the second equation we obtain the value for x_1 :

$$x_{1MIN} = 1 + \frac{1}{\gamma}(e^{-2\gamma} - 1)$$

what we have obtained are two new safety boundaries.
If we restrict our recovery system so that it keeps x_1 between these two new boundaries :

$$x_{1MIN} \leq x_1 \leq x_{1MAX}$$

we can be sure that the delays involved in the presence of rate limiters will not cause the system to exceed the controllability boundaries.
This of course restrict the area of free control of the system. The more strict are the rate limiters and the bigger will be the region lost compared to the case without rate limiters.
An initial idea could be to proceed as showed in the previous chapter with the first order system, having a controller that dynamically change the boundary according to the state of the input variables.
Known the position of the two control inputs, it is possible to know the value of the parameter x_2, and therefore it is known the value of the maximum x_1 allowed from which it is a straight forward to obtain a maximum value for z_1, beyond which a full converging control action is not able anymore to keep the system into a controllable configuration, due to the finite speed at which we can vary the input signal.
Therefore our controller may switch from normal controlling to a full recovery action if the dynamic z_1 value getting too close.
While this system works most of the times, it does not analytically assure that the recovery system always switches-on in time to save the system.
It is possible to construct some simulations to show that in some cases the limits are exceeded and the auto-recovery fails to work properly.
To better understand the problem of this approach, we may notice that not always the system is able to move the effective control signal at it maximum speed.
The maximum effective control signal is defined as :

$$\bar{\delta} = |\beta_1|\bar{\delta}_{ec} + |\beta_2|\bar{\delta}_{cc}$$

and represents the way control inputs may affect the unstable mode.

In the same way we could compute the maximum speed at which the effective control signal may change, and this is a function of the individual values of the maximum control rates of the single inputs.

$$\overline{\Delta} = |\beta_1|\Delta_{ec} + |\beta_2|\Delta_{cc}$$

Note that we may have two different values of the control inputs leading to the same value of the effective control signal and this means that not always during an evolution from one state to another of the effective control signal the maximum rate can be used.
If, for example, the canard input reaches its saturation limit, then only the elevator input will change and the maximum changing rate for the effective signal will be reduced to :

$$\overline{\Delta} = |\beta_1|\Delta_{ec}$$

in general the path between two values of the effective signal will be formed by an initial part in which both the input signals will change at their maximum rate and therefore the maximum rate for the effective control signal will be used and a terminal phase during which the evolution speed may be significantly decreased.
This additional delay may be large enough to erode the safety margin of the recovery system thus making the recovery system unable to pull the system inside the controllability edge in time.
Given a certain value of the unstable mode z_l and a position of the control signals δe and δc, we will now illustrate a procedure to compute a correct maximum value for the unstable mode before which no recovery action is needed.
During the following paragraph we will consider valid the following statement :

$$\frac{\overline{\delta}_e}{\overline{\Delta}_e} = \frac{\overline{\delta}_c}{\overline{\Delta}_c}$$

that means that the canard and the elevator inputs take the same time to reverse the control action.
If this is not true the following concept will not change qualitatively, instead there will only be different lines along which the input signals will move when changing the value of the effective control signal.
We will now focus on the upper limit of the unstable mode, that is, finding the maximum positive value at which the recovery should start.
Since positive angles of the canard deflection affect qualitatively the aircraft in the same way as negative elevator deflections, the coefficient $\beta 1$ and $\beta 2$ are characterized by opposite signs. Therefore a typical recovery response to a dangerously high value of z_l will result into δec moving to its maximum positive value and δcc moving towards its maximum negative value.
We need now to calculate the evolution of the system during the two phases to find the last possible value of z_l that allows the recovery action to keep the system between the boundaries. Such calculation will need to be done backwards, starting from the limit (reached when $x_1=1$ and $x_2=-1$) and calculating the evolution with the reduced speed until a certain value of x_2 is reached, and this will allow to know the value of x_1 in the intermediate point when one of the two inputs become saturated. The first part of the trajectory will be then computed knowing the final value that has to be reached.
In general, trajectories towards the positive limits of z_l are given by the general integral :

$$(x_1 + x_2 - \frac{1}{\gamma})e^{\gamma x_2} = C$$

whose meaning is fully explained in the previous pages.
We recall for ease the correspondence between the variables of the equation and the physical variables of the system :

$$x_1 = \frac{\lambda z_1}{\overline{\delta}}$$

$$x_2 = \frac{\delta}{\overline{\delta}}$$

$$\gamma = \frac{\lambda \overline{\delta}}{\Delta}$$

a more useful expression of the above general integral is one that express the value of x_1 as a function of x_2, γ and of a point x_{p1} and x_{p2} that we want to be included in the trajectory. This leads to the following expression :

$$x_1 = \frac{1}{\gamma} - x_2 + (x_{p1} + x_{p2} - \frac{1}{\gamma})e^{\gamma(x_{p2} - x_2)}$$

Now, we start from a given value of z_1, δ_{ec} and δ_{cc} ; it is immediately possible to calculate the value of x_2 corresponding to the actual values of δ_{ec} and δ_{cc} :

$$x_2' = \frac{\beta_1 \delta_{ec} + \beta_2 \delta_{cc}}{\overline{\delta}}$$

now we have to make distinction between three different cases :

5.10.1 Case one : $\delta_{cc} = - \delta_{ec}$

This is the simplest case, the input signals will change together until the endpoint will be reached simultaneously. In this case no special considerations are needed and everything goes like a system with a single input. We will no more spend words on this case.

5.10.2 Case two : $\delta_{cc} > - \delta_{ec}$

In this case, after an initial phase during which the two control signals will move together, the elevator input will saturate and the canard input signal will evolve alone until the end point.
During the first phase the input variables will assume the following values :

$$\delta_{ec}' \leq \delta_{ec} \leq \overline{\delta}_e$$

$$-\overline{\delta}_c + \delta'_{ec} + \delta'_{cc} \le \delta_{cc} \le \delta'_{cc}$$

While during the terminal phase, the ranges will become :

$$\delta_{ec} = \overline{\delta}_e$$

$$-\overline{\delta}_c \le \delta_{cc} \le -\overline{\delta}_c + \delta'_{ec} + \delta'_{cc}$$

We can define the intermediate point as the one at which :

$$\begin{bmatrix} \delta_{ec} \\ \delta_{cc} \end{bmatrix} = \begin{bmatrix} \delta''_{ec} \\ \delta''_{cc} \end{bmatrix} = \begin{bmatrix} \overline{\delta}_e \\ -\overline{\delta}_c + \delta'_{ec} + \delta'_{cc} \end{bmatrix}$$

and we will have a corresponding value for x_2 given by the expression :

$$x''_2 = \frac{\beta_1 \delta''_{ec} + \beta_2 \delta''_{cc}}{\overline{\delta}} = \frac{\beta_1 \overline{\delta}_e + \beta_2 (-\overline{\delta}_c + \delta'_{ec} + \delta'_{cc})}{\overline{\delta}}$$

Known the value of x_2 in the intermediate point we can now trace back the trajectory from the point (1 ; -1) with the formula shown previously :

$$x_1 = \frac{1}{\gamma} - x_2 + (x_{p1} + x_{p2} - \frac{1}{\gamma}) e^{\gamma(x_{p2} - x_2)}$$

which in this case becomes :

$$x''_1 = \frac{1}{\gamma_2} - x''_2 - \frac{1}{\gamma_2} e^{\gamma_2(-1 - x''_2)}$$

where γ_2 is calculated considering only the canard input signal varying with the following relation :

$$\gamma_2 = \frac{\lambda_1 \overline{\delta}}{|\beta_2| \overline{\Delta}_c}$$

now that both x_1 and x_2 of the intermediate point are known, we can finally know the value of x'_1 corresponding to x'_2, who is known from the beginning.
This is accomplished using again the previous formula, with the general γ and starting from the known intermediate point :

$$x'_1 = \frac{1}{\gamma} - x'_2 + (x''_1 + x''_2 - \frac{1}{\gamma}) e^{\gamma(x''_2 - x'_2)}$$

5.10.3 Case three : $\delta_{cc} < - \delta_{ec}$

This is the opposite case and the same general rules may be applied. This time the terminal phase will be performed with the elevator input changing alone while the canard input will be saturated.
Once known the value of x'_l it is a straightforward to compute the maximum allowable value of z_l and this will be compared with the actual value of z_l to decide whether is the case or not for the emergency recover to take place.

5.11 Numerical implementation

Implementing a controller with fixed boundaries in the case of a rate limited input it is a straightforward of the procedure adopted in the previous chapter.
The same recovery system implemented at the beginning of this chapter can be used, simply moving the safety limit to a lower one using the formulas introduced in the previous paragraphs.
If instead we wish to have dynamic boundaries that allow to fully use all the theoretical flight envelope available, we need a more complex controller that gets the current value of the saturated control inputs and uses them to compute in real-time the safety boundaries, with different algorithms chosen according to the situation.
In the appendix is presented a function that takes as input the position of the saturated actuators and computes the dynamic boundary ; it will be the duty of the controller to use this routine to check in any moment if the safety limit is trespassed of not.
The disadvantage of this routine is that it is quite heavy from a computational point of view, since it contains logical operators combined with exponential operators ; this means that in some cases the computational load may be excessive if the resources are not so high and this may slow down the frequency at which these safety checks are performed.
However, once known the computational algorithm, it is possible to think about simplifications in the controller to let it compute the dynamic boundary with few operations, still ensuring an acceptable error, as done in the previous chapter for first order systems.

5.12 Conclusions

Unstable multivariable systems with input saturations presented several control problems that critically influence the safety of the system. Multiple outputs give also more flexibility and allow more choices of control strategies.
The presence of rate limiters in general requires predictive actions. This means that it is critical to have a reliable model of the plant, the better the model is, the smaller is the safety margin and the larger is the operating range for the system.
Applying dynamic boundaries presents unexpected problems in the multi-input case. These problems occur when considering an independent movement of two control surfaces. Since several devices affect the position of the control surfaces, it is more general to consider their movement to be independent and analyze the problem in these terms.
Independently moving surfaces increase the complexity of the algorithm, as shown in the last paragraph, and this usually overloads the controller with heavy operations that are much more demanding than the rest of the control algorithm.
According to the precision of the model it is reasonable to have approximations to avoid the calculations of exponential functions. Even with approximations, the software will be more complex than before.
More complex algorithms usually result into less reliable software unless a lot resources are spent to formally analyze and debug the routines. When designing mission-critical

components, software reliability is a major issue that must be kept into account when choosing control strategies.
The more complex controller and the much more difficult implementation of a dynamic boundary controller is the price required to fully use the flight envelope.

6. APPENDIX

6.1 Stability derivatives of the model

The matrix elements that are characterizing the short period of the plane can be easily be estimated through the knowledge of the stability derivatives of the aircraft.
In this case we can have a satisfactorily representation of the above model trough conventional stability derivatives which can be used to investigate the effect of small changes of such coefficients.
A state space representation of the model would be :

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \\ \dot{\delta}_e \\ \dot{\delta}_c \end{bmatrix} = \begin{bmatrix} \frac{Z_w}{u_0} & \sim 1 & \sim 0 & * & * \\ M_w + M_{\dot{w}} Z_{\dot{w}} & M_q + M_{\dot{w}} u_0 & \sim 0 & * & * \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & * & 0 \\ 0 & 0 & 0 & 0 & * \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \theta \\ \delta_e \\ \delta_c \end{bmatrix} + Bu$$

where the terms are expressed by the following expressions :

$$Z_w = -\frac{(C_{L_\alpha} + C_{D_0}) Q S}{m u_0}$$

$$M_w = C_{m_\alpha} \frac{Q S \bar{c}}{u_0 I_y}$$

$$M_{\dot{w}} = C_{m_{\dot{\alpha}}} \frac{\bar{c}}{2 u_0} \frac{Q S \bar{c}}{u_0 I_y}$$

$$Z_{\dot{w}} = -C_{z_{\dot{\alpha}}} \frac{\bar{c}}{2 u_0} \frac{Q S}{u_0 m}$$

$$M_q = C_{m_q} \frac{\bar{c}}{2 u_0} \frac{Q S \bar{c}}{I_y}$$

Note that some elements that in the theoretical model are supposed to assume the value zero or one, assume in reality values lightly different.

In particular the third column of the matrix A should be composed of all zero term, while in reality has some very small non-zero terms that turn the null eigenvalue into a slightly unstable one. It has been reported that this small difference is caused by the presence of the horizontal trim tab.

6.2 MATLAB routines used in chapter 5

6.2.1 Script CANARD.M

This MATLAB script is used by all the models to extract numerical data for the simulations. It contains the numerical model and performs several other computations to store in appropriate variables the value of several other characteristics of the model itself.

```
%
% Function CANARD.M
%
A=[[-1.3936  0.9744  -0.0019  -0.5349  -0.0071];[5.687  -1.1827
0.0002 -25.9398 7.9642];[0 1 0 0 0];[0 0 0 -20 0];[0 0 0 0 -
20]]
B=[[0.0676 -0.0313];[1.527 0.4002];[0 0];[20 0];[0 20]]
C=[[1 0 0 0 0];[0 1 0 0 0];[0 0 1 0 0];[0 0 0 1 0];[0 0 0 0
1]]
D=[[0 0];[0 0];[0 0];[0 0];[0 0]]
%
% *** Elevator Transfer Function (1) ***
%
A1=A;
B1=B(:,1);
C1=C;
D1=D(:,1);
[NUM,den1]=ss2tf(A,B,C,D,1);
num1=NUM(3,:);
zeros1=roots(num1)
poles1=roots(den1)
%
% *** Canard Transfer Function (2) ***
%
A2=A;
B2=B(:,2);
C2=C;
D2=D(:,2);
[NUM,den2]=ss2tf(A,B,C,D,2);
num2=NUM(3,:);
num2(2)=0; %Put to zero an almost-zero value
zeros2=roots(num2)
poles2=roots(den2)
%
% Unstable modes
%
[V,d]=eig(A');
T=V';
TT=T(4,:);
Az=T*A*inv(T);
Bz=T*B;
%
% Feedback Matrix
```

```
%
K=place(A,B,[-20 -20 -0.05 -2.2-2.24*i -2.2+2.24*i])
%
% Sharp Stabilization (1)
%
P(3,3)=0;
P(4,4)=0.026;
P(5,5)=0;
K1=(T*B)'*P*T;
%
% Mixed Stabilisation (2)
%
P(3,3)=-0.001;
P(4,4)=0.03;
P(5,5)=0;
K2=(T*B)'*P*T;
%
% Placement of two poles (3)
%
beta(1:2,1)=Bz(4:5,1);
beta(1:2,2)=Bz(4:5,2)
alpha=beta(1,1)/beta(1,2)
alfa=[[1];[1/alpha]]
lambda(1,1)=d(4,4);
lambda(2,2)=d(5,5)
ti(1,:)=T(4,:);
ti(2,:)=T(5,:)
%kappa=place(lambda,beta*alfa,[-0.22-i*0.45 -0.22+i*0.45]);
%kappa=place(lambda,beta*alfa,[-1.5+2.01*i -1.5-2.01*i]);
%kappa=place(lambda,beta*alfa,[-2+2*i,-2-2*i]);
kappa=place(lambda,beta*alfa,[-1.5+i*2.01 -1.5-i*2.01]);
K3=alfa*kappa*ti
%
% Placement of two poles with a greater complex part (4)
%
kappa=place(lambda,beta*alfa,[-1.27+i*2.52 -1.27-i*2.52]);
K4=alfa*kappa*ti
```

6.2.2 Function CBOX.M

This is the basic controller. Meant to replace the blocks diagrams giving the same performances it only takes the two inputs and applies a linear feedback to them.

```
%
% Function CBOX.M
% Normal state feedback with a K matrix
%
function [SYS]=cbox(input);

%
% INPUTS
%
de=input(1);
dc=input(2);
x(1:5)=input(3:7);

%
% Feedback Matrix
```

```
%
K1=[[-0.1778   -0.0769    0.0003    0.0992   -0.0290];[0.0581
0.0251   -0.0001   -0.0324    0.0095]];
K2=[[-0.2306    0.0110    0.0001   -0.0255    0.0100];[0.0753
-0.0036   -0.0000    0.0083   -0.0033]];
K=K1;

%
% Feedback computation
%
feedback=(K*x')';
de=de-feedback(1);
dc=dc-feedback(2);

%
% Output
%
SYS=[de dc];
```

6.2.3 Function CBOX0.M

This function behaves like the previous one but adds a cross-feed to the inputs, replacing the corresponding block diagram.

```
%
% Function CBOX0.M
% Normal state feedback with a K matrix
% With cross feed of saturated inputs
%
function [SYS]=cbox0(input);

%
% INPUTS
%
de=input(1);
dc=input(2);
x(1:5)=input(3:7);

%
% Feedback Matrix
%
K1=[[-0.1778   -0.0769    0.0003    0.0992   -0.0290];[0.0581
0.0251   -0.0001   -0.0324    0.0095]];
K2=[[-0.2306    0.0110    0.0001   -0.0255    0.0100];[0.0753
-0.0036   -0.0000    0.0083   -0.0033]];
K=K1;

%
% Feedback computation
%
feedback=(K*x')';
de=de-feedback(1);
dc=dc-feedback(2);

%
% Cross-Feed of Saturated Control Surfaces
%
```

```
dcplus=0;
deplus=0;
if de > 30
       dcplus=-(de-30)*5;
elseif de < -30
       dcplus=-(de+30)*5;
end;
if dc > 30
       deplus=-(dc-30)*.3;
elseif dc < -30
       deplus=-(dc+30)*.3;
end;
de=de+deplus;
dc=dc+dcplus;

SYS=[de dc];
```

6.2.4 Function CBOX1.M

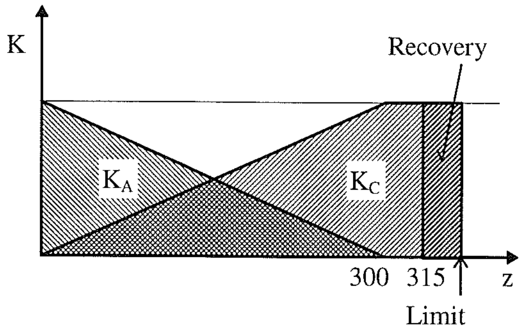
In this function the recovery logic is added with a monitoring on the z variable.

```
%
% Function CBOX1.M
% Normal state feedback with a K matrix
% Checks the unstable mode Z
% Full recovery if |Z|>limit
%
function [SYS]=cbox1(input);
limit=310;
%
% INPUTS
%
de=input(1);
dc=input(2);
x(1:5)=input(3:7);

%
% Feedback Matrix
% (Several Choices)
%
K1=[[-0.1778   -0.0769    0.0003     0.0992    -0.0290];[0.0581
0.0251   -0.0001   -0.0324     0.0095]];
K2=[[-0.2013   -0.0878   -0.0023     0.1132    -0.0331];[0.0658
0.0287    0.0007   -0.0370     0.0108]];
K3=[[-0.3399   -0.0273    0.0003     0.0230    -0.0040];[0.1110
0.0089   -0.0001   -0.0075     0.0013]];
K4=[[-0.4177   -0.0145    0.0003     0.0018     0.0033];[0.1364
0.0047   -0.0001   -0.0006    -0.0011]];
K=K2; % Matrix used

%
% Unstable mode
%
v=[0.8098 0.3502 -0.0014 -0.4518 0.1321];
z=v*x';

%
% Feedback computation
```

```
%
feedback=(K*x')';
de=de-feedback(1);
dc=dc-feedback(2);

%
%  Cross Feed of saturated inputs
%
dcplus=0;
deplus=0;
if de > 30
       dcplus=-(de-30)*5;
elseif de < -30
       dcplus=-(de+30)*5;
end

if dc > 30
       deplus=-(dc-30)*.3;
elseif dc < -30
       deplus=-(dc+30)*.3;
end
de=de+deplus;
dc=dc+dcplus;

%
% Emergency Recovery
%
if z > limit
        de=30;
        dc=-30;
end;
if z < -limit
        de=-30;
        dc=30;
end;

%
% Output
%
SYS=[de dc];
```

6.2.5 Function CBOX2.M

This function implements the simplest scheme shown of hybrid controlling.




```
%
% Function CBOX2.M
% Normal state feedback with a K matrix
% Checks the unstable mode Z
% Full recovery if |Z|>limit
% Hibrid two-state controlling
%
function [SYS]=cbox1(input);
limit=310;
%
% INPUTS
%
de=input(1);
dc=input(2);
x(1:5)=input(3:7);

%
% Feedback Matrix
% (Several Choices)
%
K1=[[-0.1778   -0.0769    0.0003    0.0992    -0.0290];[0.0581
0.0251   -0.0001   -0.0324    0.0095]];
K2=[[-0.2013   -0.0878   -0.0023    0.1132    -0.0331];[0.0658
0.0287    0.0007   -0.0370    0.0108]];
K3=[[-0.3399   -0.0273    0.0003    0.0230    -0.0040];[0.1110
0.0089   -0.0001   -0.0075    0.0013]];
K4=[[-0.4177   -0.0145    0.0003    0.0018     0.0033];[0.1364
0.0047   -0.0001   -0.0006   -0.0011]];


%
% Unstable mode
%
v=[0.8098 0.3502 -0.0014 -0.4518 0.1321];
z=v*x';

%
% Matrix Combination
%
K=((300-abs(z))/300)*K4+(abs(z)/300)*K1;
if abs(z)>300
     K=K1;
end;
```

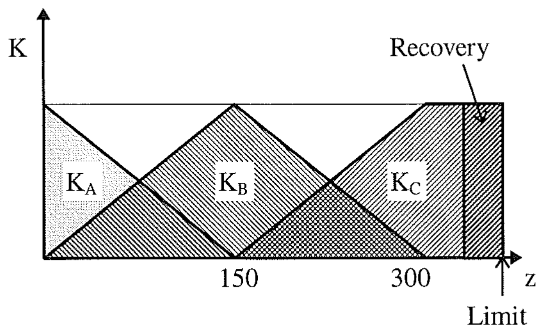
```
%
% Feedback computation
%
feedback=(K*x')';
de=de-feedback(1);
dc=dc-feedback(2);

%
%  Cross Feed of saturated inputs
%
dcplus=0;
deplus=0;
if de > 30
       dcplus=-(de-30)*5;
elseif de < -30
       dcplus=-(de+30)*5;
end

if dc > 30
       deplus=-(dc-30)*.3;
elseif dc < -30
       deplus=-(dc+30)*.3;
end
de=de+deplus;
dc=dc+dcplus;

%
% Emergency Recovery
%
if z > limit
        de=30;
        dc=-30;
end;
if z < -limit
        de=-30;
        dc=30;
end;

%
% Output
%
SYS=[de dc];
```

6.2.6 Function CBOX3.M

In this second case of Hybrid Control the matrix combination block chooses between three different matrices and differs from the previous function in the following lines :

```
%
% Matrix Combination
%
if abs(z) < 150
     K=((150-abs(z))/150)*K4+(abs(z)/150)*K3;
elseif abs(z) < 300
     K=((300-abs(z))/150)*K3+((abs(z)-150)/150)*K1;
else
     K=K1;
```

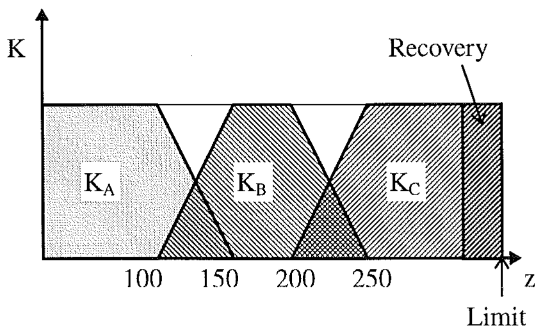
```
end;
```



6.2.7 Function CBOX4.M

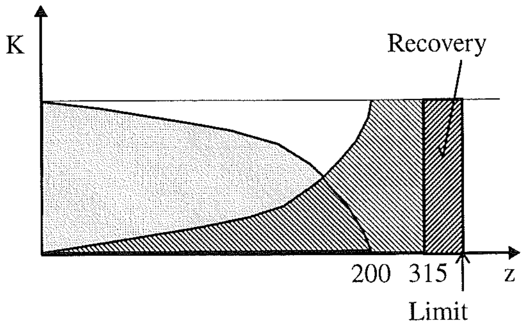
This time plateaus are inserted with the use of the following code :

```
%
% Matrix Combination
%
if abs(z) < 100
     K=K4;
elseif abs(z) < 150
     K=((150-abs(z))/50)*K4+((abs(z)-100)/50)*K3;
elseif abs(z) < 200
     K=K3;
elseif abs(z) < 250
     K=((250-abs(z))/50)*K3+((abs(z)-200)/50)*K1;
else
     K=K1;
end;
```



6.2.8 Function CBOX5.M

In this function the use of parabolic membership function is introduced.



Here is the corresponding code that decides the weight of the two matrices :

```
%
% Function CBOX5.M
% Normal state feedback with a K matrix
% Checks the unstable mode Z
% Full recovery if |Z|>limit
% Hibrid three-state fuzzy controlling
%
function [SYS]=cbox1(input);
limit=310;
%
% INPUTS
%
de=input(1);
dc=input(2);
x(1:5)=input(3:7);

%
% Feedback Matrix
% (Several Choices)
%
K1=[[-0.1778   -0.0769   0.0003   0.0992   -0.0290];[0.0581
0.0251   -0.0001   -0.0324   0.0095]];
K2=[[-0.2013   -0.0878   -0.0023   0.1132   -0.0331];[0.0658
0.0287   0.0007   -0.0370   0.0108]];
K3=[[-0.3399   -0.0273   0.0003   0.0230   -0.0040];[0.1110
0.0089   -0.0001   -0.0075   0.0013]];
K4=[[-0.4177   -0.0145   0.0003   0.0018   0.0033];[0.1364
0.0047   -0.0001   -0.0006   -0.0011]];


%
% Unstable mode
%
v=[0.8098 0.3502 -0.0014 -0.4518 0.1321];
z=v*x';

%
% Matrix Combination
%
if abs(z) < 200
     K=K4*((200^2-abs(z)^2)/200^2)+K1*(abs(z)^2)/(200^2);
else
     K=K1;
```

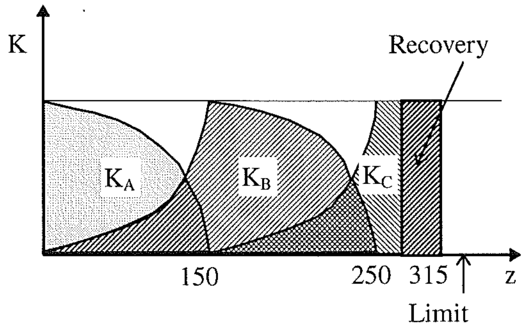
```
end;

%
% Feedback computation
%
feedback=(K*x')';
de=de-feedback(1);
dc=dc-feedback(2);

%
%  Cross Feed of saturated inputs
%
dcplus=0;
deplus=0;
if de > 30
       dcplus=-(de-30)*5;
elseif de < -30
       dcplus=-(de+30)*5;
end

if dc > 30
       deplus=-(dc-30)*.3;
elseif dc < -30
       deplus=-(dc+30)*.3;
end
de=de+deplus;
dc=dc+dcplus;

%
% Emergency Recovery
%
if z > limit
        de=30;
        dc=-30;
end;
if z < -limit
        de=-30;
        dc=30;
end;

%
% Output
%
SYS=[de dc];
```

6.2.9 Function CBOX6.M

In this last control box, three function are mixed with parabolic membership functions using the following code variation :

```
%
% Matrix Combination
%
if abs(z) < 150
     K=K4*((150^2-abs(z)^2)/150^2)+K3*(abs(z)^2)/(150^2);
elseif abs(z) < 200
     K=K3;
elseif abs(z) < 250
```

```
      K=K3*((250^2-abs(z)^2)/(250^2-200^2))+K1*((abs(z)^2-
200^2)/(250^2-200^2));
else
      K=K1;
end
```



6.2.10 Function STOP.M

This function allows the realization of a dynamic boundaries controller ; it takes as inputs the position of the control surfaces and return the maximum controllable value of z.
It makes uses of a small function called "margin" which is reported thereafter.

```
%
% STOP.M
% Dynamic boundaries
% as a function of the control surfaces position
%
Function z = stop(de,dc);
%
% Numerical Constants
%
lambda=1.0662;
demax=30;
dcmax=30;
Demax=300;
Dcmax=300;
beta1=-8.4471;
beta2=2.7576;
%
% Maximum values
%
dmax=abs(beta1)*demax+abs(beta2)*dcmax
Dmax=abs(beta1)*Demax+abs(beta2)*Dcmax
Dmax1=abs(beta1)*Demax
Dmax2=abs(beta2)*Dcmax
%
% Different values of Gamma
% according to the state of saturation of the surfaces
%
gamma=lambda*dmax/Dmax
gamma1=lambda*dmax/Dmax1
gamma2=lambda*dmax/Dmax2
%
% Discrimination between different cases
% and computation of the appropriate margin
% using margin.m as a sub-function.
```

```
%
if dc > -de
      x2a=(beta1*de+beta2*dc)/dmax
      x2b=(beta1*demax+beta2*(-dcmax+de+dc))/dmax
      x1b=margin(gamma2,x2b,1,-1)
      x1a=margin(gamma,x2a,x1b,x2b)
else
      x2a=(beta1*de+beta2*dc)/dmax
      x2b=(beta1*(demax+de+dc)+beta2*(-dcmax))/dmax
      x1b=margin(gamma1,x2b,1,-1)
      x1a=margin(gamma,x2a,x1b,x2b)
end;
%
% Output
%
z = x1a
```

6.2.11 Function MARGIN.M

Computes the operating boundary given the values of the variables γ, x_{2a}, x_{1b}, x_{2b}. It is called by MARGIN.M

```
%
% Function MARGIN.M (Used by STOP.M)
% X1=margin(gamma,X2,Xp1,Xp2)
%
function x1 = margin(gamma,x2,xp1,xp2);
x1=(1/gamma)-x2+(xp1+xp2-(1/gamma))*exp(gamma*(xp2-x2));
```

REFERENCES

[1] Karl Joan Åström, *Reglerteori*, Almqvist & Wiksell, Stockholm 1968

[2] Torkel Glad, *Stabilitetsområde for flygplanmodel,* Preliminary version 940628, Reglerteknik, ISY, LiTH, Linköping.

[3] John H. Blakelock, *Automatic Control of Aircrafts and Missiles*, Wiley & Sons 1965

[4] Karl Joan Åström, Tore Hägglund, *PID Controllers : Theory, Design and Tuning*, 2nd Edition. Instrument Society of America, 1995

[5] Robert C. Nelson, *Flight Stability and Automatic Control,* McGraw-Hill, 1989

[6] Edward Seckel, *Stability and Control of Airplanes and Helicopters*, Academic Press, New York, 1964.

[7] Rundqwist L., *Rate limiters with phase compensation in JAS-39 Gripen*, Proceedings of XI Hungarian Days of Aeronautical Sciences, Budapest, Hungary, June 5-7, 1996.

[8] Rundqwist L. and Hillgren R., *Phase Compensation of Rate Limiters in JAS-39 Gripen*, AIAA paper 96-3368, 1996 AIAA Atmospheric Flight Mechanics Conference, San Diego, CA, July 29-31 1996.

[9] Rundqwist L., *Rate limiters with phase compensation*, proceedings of 20th Congress if ICAS, Sorrento, Italy, September 9-13 1996.

[10] Rundqwist L. and Ståhl-Gunnarsson, *Phase Compensation of Rate Limiters in Unstable Aircraft*, proceedings of 1996 IEEE Conference on Control Applications, Dearborn, MI, September 15-18 1996.

[11] Timothy J. Ross, Fuzzy Logic with Engineering Applications, McGraw-Hill, 1985