

HIGH PERFORMANCE PI(D) CONTROLLER ON SHIP STEERING WITH A CONDITIONAL RESET WINDUP

R. Ferreiro Garcia (1), C. Antonio F. Ameal (2)

(1) Dept. of Ind. Engineering, (2) Dept. of Maritime Sciences
E.S Marina Civil, Paseo de Ronda 51- 15011. E-mail:ferreiro@udc.es
University of A Coruna.Spain

Abstract: In this work it is described some practical aspects of PI(D) controllers regarding to conditional integration as an alternative to classical reset windup. This contribution concerns to the integral action, which is managed as function of control error sign. The effect due to the action of reset to zero or clear integral action when sign error changes, is a drastic reduction in response overshoot to command inputs for a special type of common industrial control systems among which, are ship steering control. Such strategy take advantages over conventional reset windup of PI(D) controllers in some applications such as feedback control of second order processes with at least, one of the dominant roots located at the origin of complex plane.
Copyright () 2001 IFAC

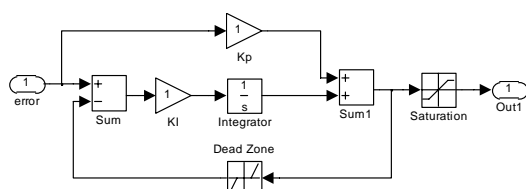
Keywords: Reset windup, conditional integration, clearing integral action, overshoot suppression,

1. INTRODUCTION TO CONDITIONAL INTEGRATION AND ANTIRESET WINDUP.

Some practical aspects of PI(D) controllers deal with non-linear constraints inherent to the basic control actions. When a controller with integral action, PI or PID sees an error signal for a long period of time, it integrates the error until it reaches a maximum, usually 100% of scale or a minimum, 0% of scale (Astrom and Wittenmark, 1984; Shinskey 1996). This is called reset windup. A sustained error signal can occur for a number of reasons, but the use of override control is one major cause. If the main controller has integral action, it will windup when the override controller has control of the final control element, for instance a valve. If the override controller is a PI controller, it will windup when the normal controller is setting the valve. For this reason, such reset windup problem must be recognised and solved. Figure 1 shows a PI controller modelled in Simulink to implement a integration reset windup.

Fig. 1. An alternative to implement an integration reset windup

The strategy depends on the controller hardware and software used. In pneumatic controllers reset windup can be prevented by using external reset feedback, by means of feeding back the signal of the control valve to the reset chamber of the controller instead of the controller output (Luyben, 1989, Shinskey, 1996). This lets the controller integrate the error when its output is going to the valve, but breaks the integration loop when the override controller is setting the valve. In analog electronics it is used the same strategy. Figure 2 shows a PI controller modelled in Simulink to implement a integration reset windup by means of a non linearity.



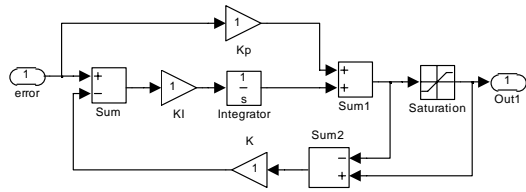


Fig. 2. Implementation of a reset windup with a non-linearity

In computer control and software based microsystems the integration action is turned off when the controller does not have control of the valve or the associated final control element. In conclusion, a regulator with integral action is an input output unstable system. Its unstable mode can give rise to difficulties under certain circumstances. Reset windup or integrator saturation can occur if the output saturates and the controller continues to integrate the error. The output of the integrator can then assume very large values, and it can take a long time to get it back to a normal value again. This problem is automatically avoided when the velocity form of the algorithm is used, because the integration stops automatically. When the output is limited. If it is desirable to use the position form, then some precaution must be taken. One way is to stop updating the integrator when the output is limited. Another way is to use conditional integration, in which the integral part of the controller is used only when the error is sufficiently small. In practical situations, the output is limited. Then, the integral part is reset to a value that gives low limit or high limit values respectively.

2. AN ALTERNATIVE TO RESET WINDUP: CI ALGORITHM

In the case of reset windup controllers, the output of the integrator can then assume only normal values restricted to high and low limits. In process control it is common the case in which such a controller is operating into the high or low limits when the error approach or becomes zero or is changing from negative to positive or from positive to negative. In that cases it is necessary certain time to get the controller action in the right direction due to the time to reduce the integral value from the actual limit to zero value. This problem is automatically avoided by setting the integral value to zero, that means the actual integration value when error sign is changing will be set to zero only once per sign change. This task will be called Clear Integration (CI) algorithm

That basic idea consists in clear the integral action value at any instant for which error sign

changes, (Ferreiro, 2001). The effect of such strategy is that the time necessary to decrease the integration value is set to zero, and consequently, controller efficiency is better in some practical cases.

Several alternative CI algorithms may be implemented. The algorithms depend on the method used to perform the integral action but in all of them the following steps must be performed

- Error sign change detection
- Reset integration value to zero

Classical software based algorithm

```

Loop until stop command is detected
get actual error value
    If sign of past cycle error = sign
of actual cycle error then continue
integration algorithm
    else set integration value to zero
(clear integration)
    end if
set old error = actual error
end loop

```

Object oriented software-based algorithm

Object oriented programming is the most common tool to implement computer control algorithms and as such, a virtual engineering environment is a facility to realise an integration action with zero reset option when error sign changes, as shown in figure 3. The else option of the if then block performs the task of clearing integral action on the accumulator block, which integrates the error.

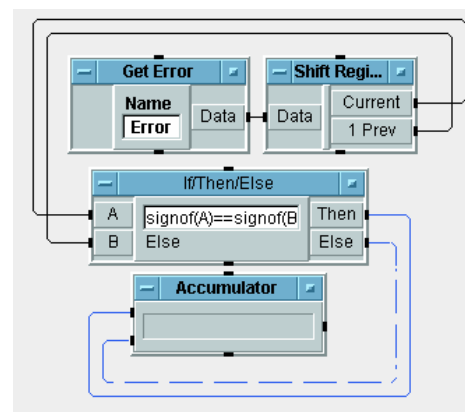


Fig.3. A software based algorithm to clear integration action implemented on a virtual engineer environment

Simulation environment software based algorithm

Implementation of proposed CI algorithm might be carried out by means of Simulink. In

figure 4 it is shown a PI controller equipped with such CI algorithm. The integrator block is specially useful to perform conditional integration under error change of sign.

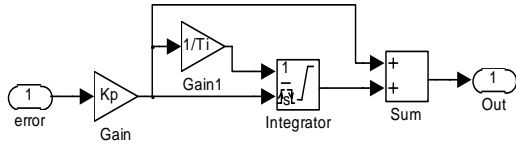


Fig. 4. Simulink implementation to clear integration action in a PI controller

Typical applications

Candidate applications to proposed controller are all second order processes whose dominant roots consists in a pair, where at least one of them is located at the origin. Location of the dominant roots in s plane and the feedback control scheme are shown in figure 5.

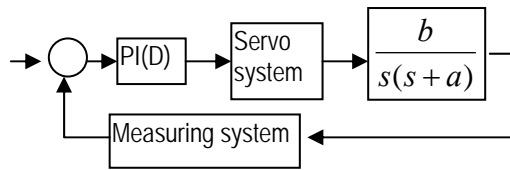


Fig. 5. Block diagram and dominant roots for system: $G(s) = \frac{b}{s(s+a)}$.

The strategy based in conditional integration by CI or clearing integral action under changes of sign of the error, becomes especially useful to reduce drastically or avoid excessive response overshoot to set point changes or command inputs. Such strategy might be used continuously or be alternatively activated when command changes are detected, being cancelled when steady state is reached. Among the processes described by mentioned models are:

- ◆ Ship positioning
- ◆ Ship steering
- ◆ Ship propulsion control

In next section it will be shown some preliminary results achieved by simulation.

3. VALIDATION OF CI ALGORITHM BY SIMULATION OF A SHIP MODEL

This section is dedicated to validate the proposed contribution. Validation of CI algorithm will be carried out on the basis of simulation results. A useful model for ship steering control purposes is proposed. It belongs to a 55 m. length minesweeper whose characteristic parameters have been researched by (Astrom and Wittenmark, 1989).

$$\begin{aligned} \frac{dv}{dt} &= \frac{u}{l} a_{11} v + u a_{12} r + \frac{u^2}{l} b_1 \delta \\ \frac{dr}{dt} &= \frac{u}{l^2} a_{21} v + u a_{22} r + \frac{u^2}{l^2} b_2 \delta \\ \frac{d\theta}{dt} &= r \end{aligned} \quad (1)$$

The non linear model, as it is originally achieved, is shown in figure 6. In this general model a servo system or a steering gear servo is included. The purpose of the simulation is to achieve graphic time response curves in order to be compared by visual inspection. A criterion to validate responses concerns to the capacity to track a reference signal or set point change, with rapid following and accurate characteristics. Comparison of the ship responses to command inputs deals with two different controller types: when autopilot controller is equipped with a CI algorithm versus classic reset windup.

4. MAIN RESULTS AND CONCLUSIONS

An exhaustive series of test have been carried out on proposed ship model. The total amount of tests related to such cases respond with a clear advantageous difference, regarding conventional reset windup: overshoot is drastically reduced or avoided in practical terms.

Simulation results presented by means of figures 6,7, 8, and 9 shows that for all studied cases, performance of ship steering with CI algorithm is better than using only conventional reset windup PI(D) controllers. The most relevant conclusions regarding ship following to step heading changes, are:

- ◆ For all tests, steering performance using CI algorithm, is better reducing overshoot.
- ◆ Using CI algorithm, reset windup might be avoided because excess integration is limited by PID output limiter, which must be applied in any case.

- ◆ The possibility of apply the PI(D) controller in different processes requires the implementation of a selector to apply conventional reset windup or CI algorithm.

REFERENCES

- Astrom K.J. and B. Wittenmark (1984). *Computer controlled systems* Ed. Prentice-Hall Inc, Englewood Cliffs, N.J.07631. USA Chap. 8. pp. 181-186
- Astrom K.J. and B. Wittenmark (1989). *Adaptive Control*. Ed. Addison-Wesley Publishing. Chap 10, pp. 355-359.
- Ferreiro, R. (2001). IEEE Transactions on Control Systems *High performance PI(D) controllers using an alternative to reset windup*. In revision phase
- Luyben, W.L. (1989). Process modeling, simulation, and control for chemical engineers Ed. McGraw-Hill, Inc. USA Chap. 8, pp. 259-262
- Shinskey,F.G. (1996). *Proces control systems. Applications, design and tuning*. Ed.McGraw-Hil, Inc. U.S.A. Chap. 4. pp. 143-150.

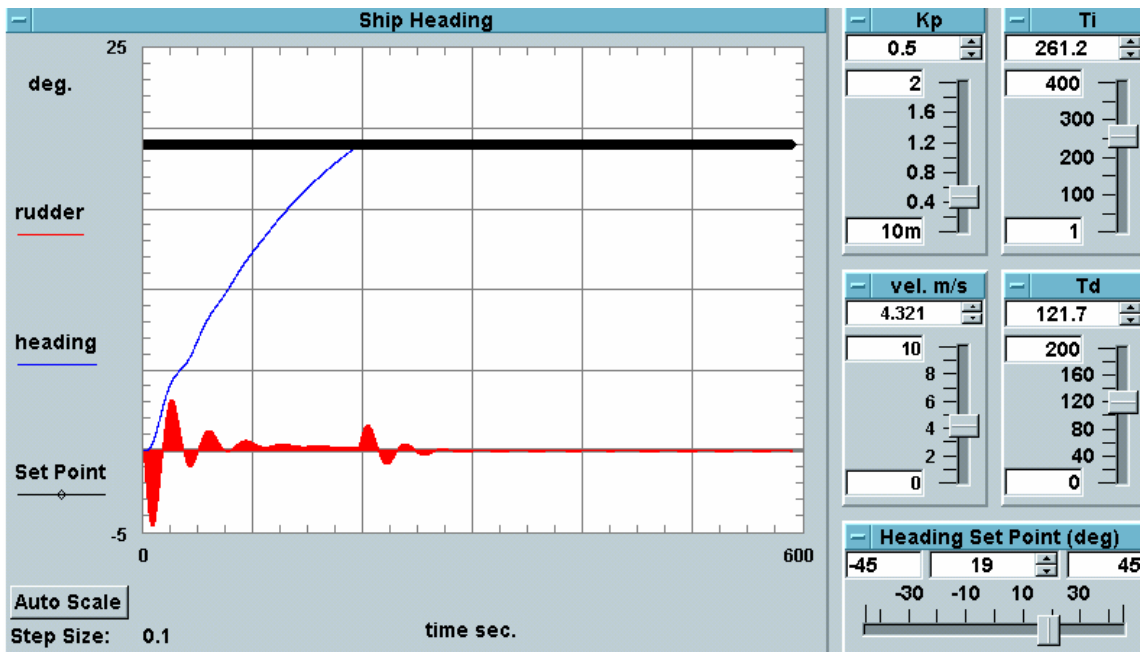


Fig. 6. The response to a 19 deg. command input with CI algorithm at 4.3 m/s velocity.

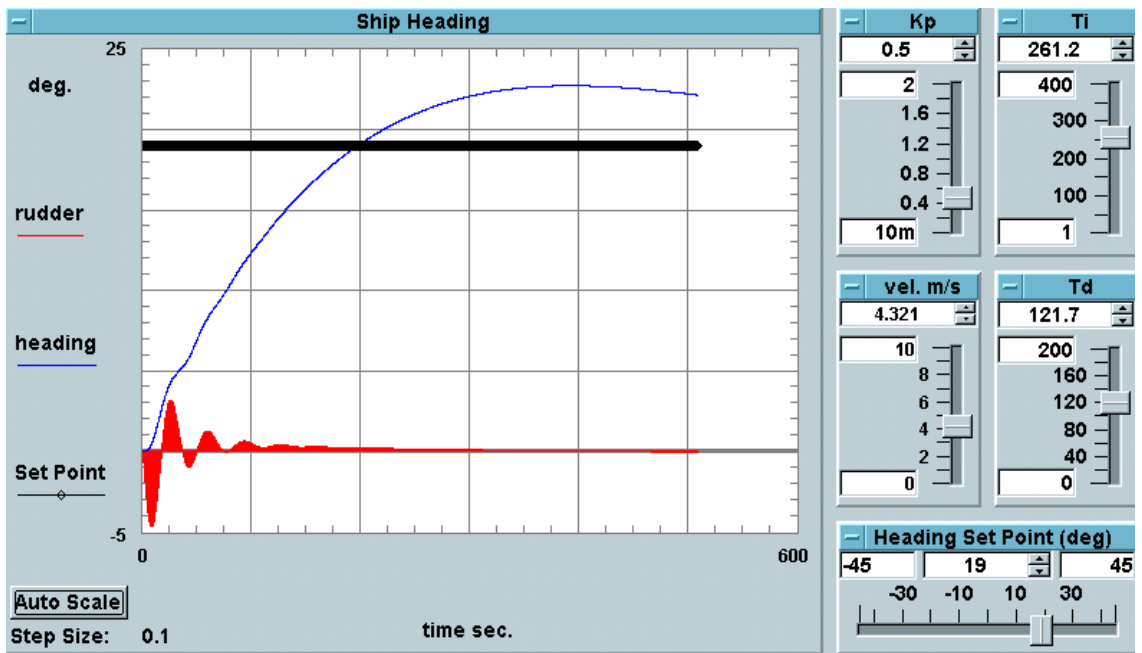


Fig. 7. The response to a 19 deg. command input with reset windup at 4.3 m/s velocity

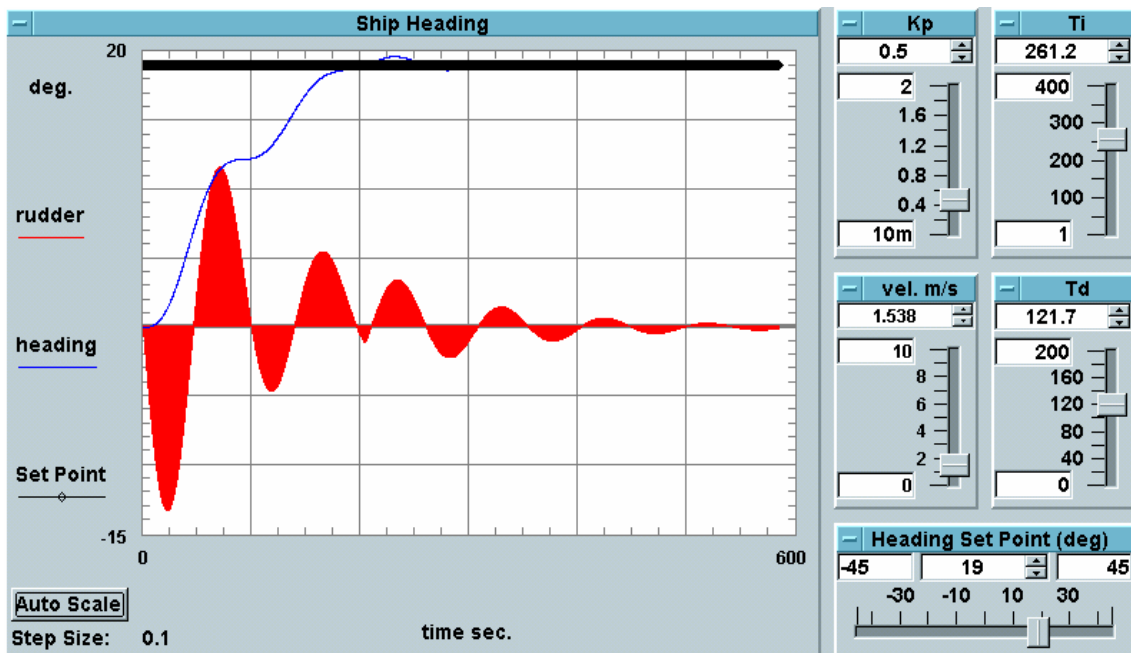


Fig. 8. The response to a 19 deg. command input with CI algorithm at 1.5 m/s velocity

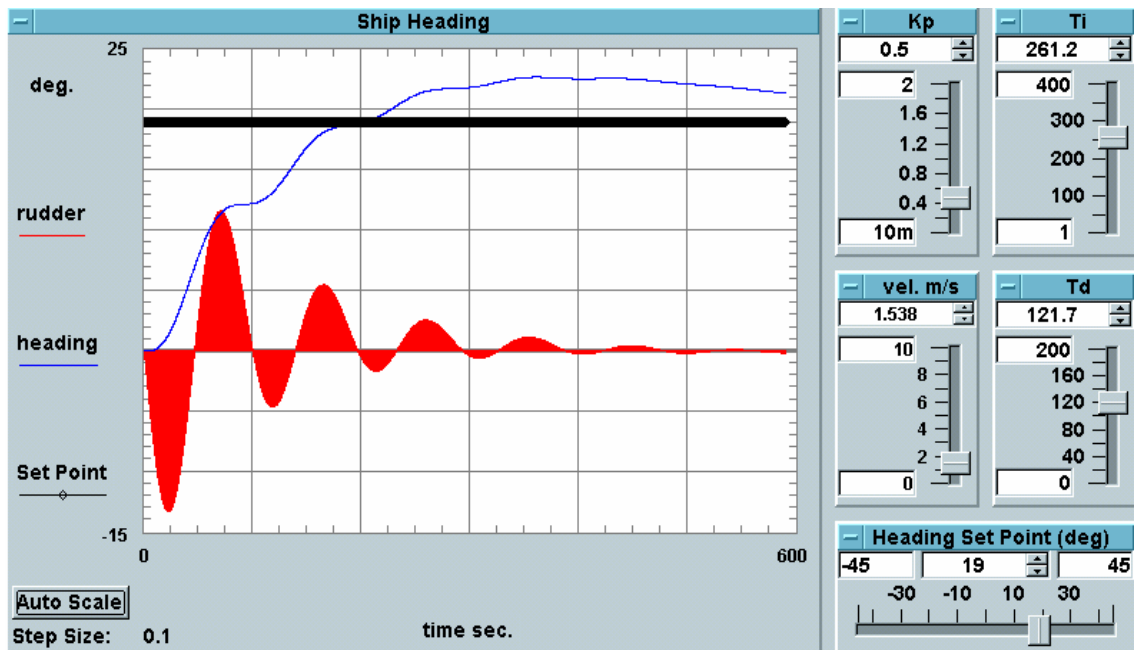


Fig. 9. The response to a 19 deg. command input with reset windup at 1.5 m/s velocity.