A MODIFIED PI-ALGORITHM FOR A GLASSHOUSE HEATING SYSTEM

A.J Udink ten Cate
Dept. of Physics and Meteorology
Agricultural University
Wageningen - The Netherlands

J. van Zeeland

Abstract

In a glasshouse heating control loop, usually saturations are inherently present. When a PI-algorithm is used for an accurate control, an anti-windup procedure has to be employed. In this paper such a procedure is presented for a typical glasshouse heating control loop, consisting of a master-slave configuration.

1. Introduction

In a typical situation in the Netherlands a glasshouse is heated by hot-water pipes of which the (water) temperature is regulated by a three-way mixing valve, mixing return water from the heating pipes with water from the main boiler. In order to obtain an accurate control of the glasshouse air temperature for this control loop a PI-type (proportional plus integral) controller is a suitable choice. In conventional analogue controllers the necessary large time constant of the integrating term of the controller may not be easily realizable in electrical components, but when a glasshouse computer is used this is easily computed digitally.

However, in the control loop typical saturations are present, which deteriorate the performance of the controller. Notably the integrating term of the PI controller can go to infinity when because of a saturation (e.g. the heat demand of the glasshouse cannot be supplied by the main boiler) the error between desired and realized glasshouse temperature cannot be minimized. To prevent this phenomenon special "anti-windup" circuits or procedures are included in the conventional controller or the computer control algorithms respectively. The commonly used anti-windup method is to straightforwardly prevent the integrating term to go outside a prespecified range. In computer control however, other anti-windup procedures are feasible. In Udink ten Cate and van de Vooren (1977) a variant of the usual procedure was used.

This paper discusses a new type of procedure for the typical glasshouse heating control loop with a master-slave configuration (fig. 2), which is inspired by a method of Hanus (1980). The algorithm is presented after a description of a typical glasshouse control problem. Simulation results are presented and the new method is shown to compare favourably with the usual method.

2. A modified algorithm

2.1 The problem

In fig. 1 a typical glasshouse control problem is depicted, which was measured during the tuning of a control algorithm in the climate glasshouse of the Glasshouse Crops Experiment Station at Naaldwijk. The heating system is of the conventional type, with hot-water heating pipes of which the water temperature is regulated by a three-way mixing-valve. The realized and desired values of the air temperature inside the glasshouse T_g are shown as well as those of the temperature of the heating pipes T_p. Evidently, the desired T_p deviates significantly from the realized T_p, which is undesirable since T_p is the actuating quantity for the control of T_g (see fig. 2.). The desired T_p is limited by a preset minimal and maximal value.

In the morning during the warming up period, the central boiler cannot meet the heat demand, so that T_g will only slowly reach its desired value (or setpoint). In control terminology a saturation occurs in the actuating signal T_p of the heating process. As soon this is accomplished the desired T_p will decrease, but T_p itself will not. This can result in a considerable overshoot of approx. 3^0C, which is usually flattened by opening of the ventilation windows, causing undesirable energy-loss.

At the end of the day the desired T_g decreases for a lower night-temperature and a similar effect takes place, now causing an undershoot. The saturation is here caused by the fact that the water temperature in the heating pipes has to decrease via heat transport into the glasshouse. In fact the behaviour of T_p with respect to the mixing valve aperture is a-symmetrical: fast response for temperature increase, slow for decrease.

Both situations as described above are undesirable. A way to circumvent the problems is to set the minimum and maximum T_p in such manner that the overshoot and undershoot are less severe. Also the changes of the desired T_g can be made more gradually. It turns out that all these measures seldom lead to the desired result, due to the strongly varying glasshouse dynamics and the interaction between the various settings. Generally, it can be remarked that the gain in the control loop is set too low in order to obtain a satisfactory result for changes of the desired T_g (setpoint changes) at the expense of a less efficient suppression of the disturbances which are acting upon the control loop.

2.2 Discrete PI algorithms

A conventional analogue PI controller with input signal e(t) the error signal, and with output signal u(t) is described by

$$u(t) = K_r\left(e(t) + \frac{1}{\tau_i}\int_0^t e(t^*)\, dt^* \right) \quad , e(0) = 0 \qquad (1)$$

where t^* is a dummy variable. In the discrete time domain a similar form can be written

$$u(k) = K_p\, e(k) + K_i \sum_{j=0}^{k-1} e(k-j) \qquad (2)$$

where k denotes the k-th sampling time and t= kT, T being the sampling interval. See Takahashi et al (1969) for a treatment of discrete control. Rewriting eqn. (2) in a separate proportional and integral form yields:

$$P(k) = K_p\, e(k) \tag{3a}$$

$$I(k) = K_i \sum_{j=0}^{k-1} e(k-j) = K_i \sum_{j=1}^{k-1} e(k-j) + K_i\, e(k) = I(k-1) + K_i\, e(k) \tag{3b}$$

$$u(k) = P(k) + I(k) \tag{3c}$$

Here especially eqn. (3b) is of interest because here I(k) is written in recursive form. Also u(k) can be written in recursive form

$$u(k) = u(k-1) + \Delta u(k) \tag{4}$$

where the form $\Delta u(k)$ is given as

$$\Delta P(k) = K_p\, (e(k) - e(k-1)) \tag{5a}$$

$$\Delta I(k) = K_i\, e(k) \tag{5b}$$

$$\Delta u(k) = \Delta P(k) + \Delta I(k) \tag{5c}$$

In fact eqn. (5c) denotes a velocity algorithm, whereas eqn. (4) is a position algorithm. Both algorithms are similar when no disturbances are present, but since eqn. (5c) only gives a change of the actuator signal, a slow drift can occur so that in a glasshouse application eqn. (4) is preferred.

When a saturation occurs in the control loop, the value of the integrating term I(k) in eqn. (3b) or equivalently (5b) can go to infinity. In a analogue controller the integral term is always clamped at the nominal value of the power supply but in a computer explicit measures have to be taken by introducing an anti-windup procedure. Usually I(k) is limited within a preset range, but also other procedures can be worked out, see Udink ten Cate and van de Vooren (1977) for a glasshouse heating control problem.

2.3 A modification

Recently Hanus (1980) has suggested a new anti-windup procedure by changing eqn. (4) into

$$u(k) = u_r(k) + \Delta u(k) \tag{6}$$

where $u_r(k)$ is the realized actuator output. This implies that the saturation must occur in the actuator circuit and that the actuator output is measurable. Since $\Delta u(k)$ is not limited, the momentaneous behaviour of the controller still remains of the PI-type but the magnitude of u(k) is preserved from diverging too much from the realized value $u_r(k)$.

The algorithm of eqn. (6) cannot straightforwardly be applied in a glasshouse heating control system because here a master-slave configuration is used (fig. 2). The master is a PI control algorithm to which an anti-rewind procedure is to be included. The slave controls T_p which can be seen as the actuating signal of the glasshouse. In terms of eqn. (6) $T_p \equiv u_r$. By using eqn. (6) fluctuations in T_p are not reduced (by the slave loop) but are instead used to generate a new controller output u(k), causing a drift in T_p and a subsequent poor control. In order to solve this problem an essential modification leads to the algorithm:

$$u(k) = u(k-1) + \Delta u(k) \tag{7a}$$

$$u_r(k-1) - c \leqslant u(k-1) \leqslant u_r(k-1) + c \tag{7b}$$

where it is recalled that $u_r(k) \equiv T_p(k)$ in fig. 2. Eqn. (7) means that the output of the master controller u(k) is free to move between limits imposed by $u_r(k)$, the realized value of u(k); reason to call this concep the dog-lead method. The value of c is selected to be say 5^0C. In a non-saturated situation $u_r \approx u$ and since deviations caused by fluctuations are smaller than $\|c\|$ the slave can perform its task as in a linear control loop. When a saturation occurs however the eqn. (7) prohibits a too large deviation between u and u_r. In a saturated situation fluctuations in u_r will not cause a drift since saturation only occurs in case of large variations of T_g so that this will not lead to problems The value of c is selected by trial and error methods.

2.4 Proportional kick

Because the value of u(k) is always near $u_r(k)$ the algorithms (7) and even more (6) are sensitive to proportional kick. This phenomen is best described by fig. 3. In a saturated situation the desired value of the controller -in the glasshouse the setpoint of T_g - can be well below the actual value of T_g. Imagine for example a hot summerday. When the setpoint of T_g changes in the control algorithm this is equivalent to a change of the actuale value of T_g and the control algorithm will give a reaction, see eqn. (5), thereby trading off $\Delta P(k)$ and $\Delta I(k)$ This can lead to a rise of the control signal u(k) regardless of the fact wheather the new setpoint is above or below the actual T_g. In a linear controller this behaviour is correct but in the saturated case this means opening of a heating (mixing) valve in the middle of the summer, which is not exactly desirable. This behaviour is called proportional kick since it is caused by the proportional term of the controller. In the case at hand the problem is solved by explicit cancelling off the proportional kick for positive setpoint changes which remain under the actual realized value. In the algorithm a few conditiona (IF) statements are included. Also for negative setpoint changes a simila situation can occur, but as yet this is not seen nor reported to cause problems. It can be solved in the same way.

3. Simulation

The dog-lead algorithm of eqn. (7) has been simulated and compared with a conventional type of eqn. (4). The glasshouse was simulated by a tranfer function

$$H_g = \frac{0.25\, e^{-\tau_d s}}{\tau_g s + 1}$$

with τ_g = 20 minutes and τ_d = 5 min. The operating point was defined by $T_p = 30\ ^0$C and $T_g = 15\ ^0$C . The mixing valve was described as

$$T_p = (1-b)\ T_r + b\ T_f \qquad 0 \leqslant b \leqslant 1$$

with b the aperture of the mixing valve, T_f the feedwater temperature from the main boiler and the return-water temperature:

$$T_r = \frac{T_p - T_a}{\tau_p s + 1} e^{-\tau_b s}$$

$$T_a = a(T_p - T_g)$$

with τ_p=2 min., τ_b =6 min. and a =0.05 a parameter representing the cooling of the water in the pipe. The values of the measured variables are represented in integers in units of .1 ^oC, as is commonly done in practice. In fig. 4 a situation is simulated similar to the situation in fig. 1. The desired T_g is stepwise changed from 15 ^oC to 20 ^oC and the T_f= 50 ^oC is insufficient to give a good response. The algorithm was tuned for T_f= 80 ^oC so windup will occur. A disturbance on T_p of 10 ^oC occurs at t= 90 min. to t= 210 min., the time T_g desired is put to 15 ^oC again. It is seen from fig. 4a that the modified (dog-lead) PI algorithm performs significantly better than a conventional one, which can be explained from fig. 4b and 4c. The min. and max. values of the integrating term as well as the output of the master controller were 10 ^oC and 80 ^oC respectively. In the conventional PI these values clamp the T_p (or T_{pipe}) in an anti-rewind procedure.

Also, in fig. 3 the effect of reduction of the proportional kick is investigated in simulation. It is seen that the explicit reduction is effective.

4. Conclusions

A modified algorithm is presented for the control of a glasshouse heating system. This algorithm features a significant improvement of the control in case saturations are present in the control loop, which is the case in a typical heating system in the Netherlands. The modification consists of a new anti-windup procedure in a PI-type controller which is the master in a master-slave control configuration. It is seen that an explicit reduction of proportional kick in case of (positive) setpoint changes is essential in this algorithm. In simulation the advantages of the new method are demonstrated. Presently the method is investigated in field trials and so far no problems have arisen (Dec. 1980).

References

Hanus, R., 1980. A new technique for preventing control windup, Journal A, 21(1): 15 -20.

Takahashi, Y., Rabins, J.R. and Auslander, D.M., 1969. Control and dynamic systems, Ch. 11, Addison-Wesley.

Udink ten Cate, A.J. and van de Vooren, J., 1977. Digital adaptive control of a glasshouse heating system. Proc. IFAC Conf. on Digital Computer Applications to Process Control, van Nauta Lemke (ed.), North-Holland : 505 -512.

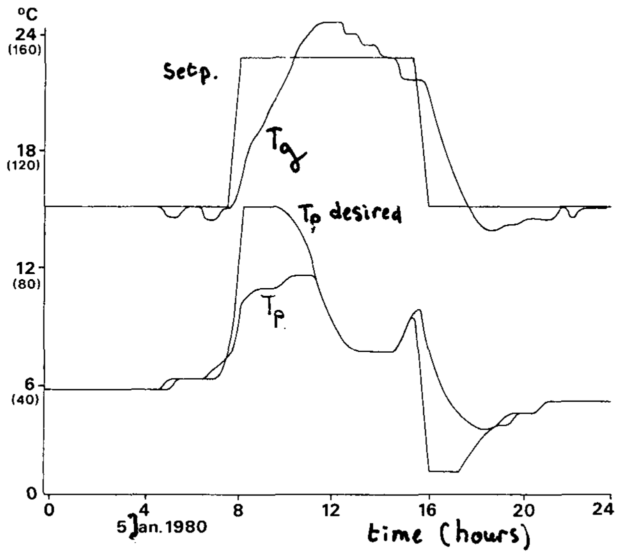

Fig. 1. Desired and actual values of glasshouse air (T_g) and heating pipe (T_p) temperatures.

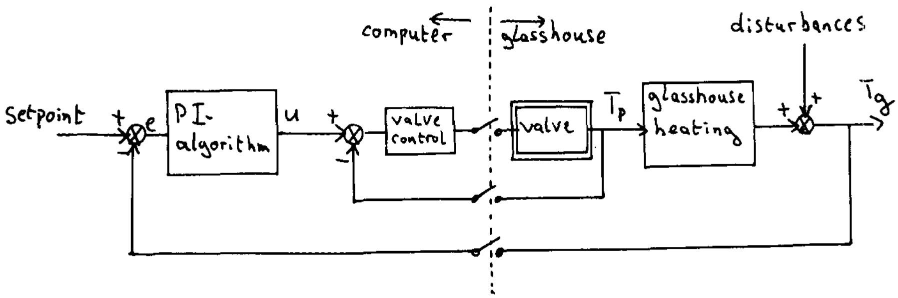

Fig. 2. Glasshouse heating system control loops.

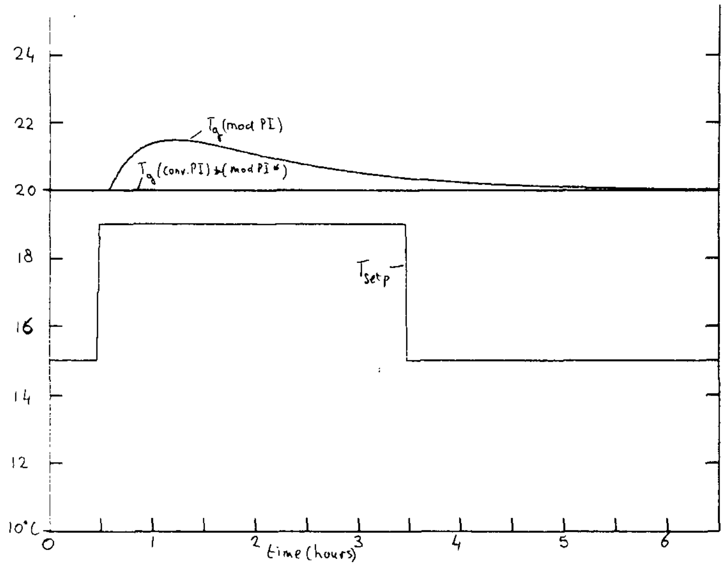

Fig. 3. Proportional kick.

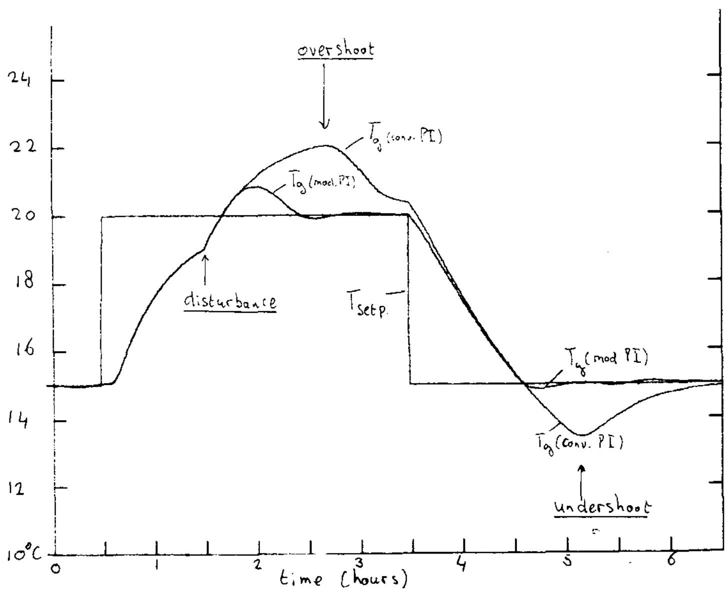

Fig. 4a. Comparison of control algorithms in simulation.

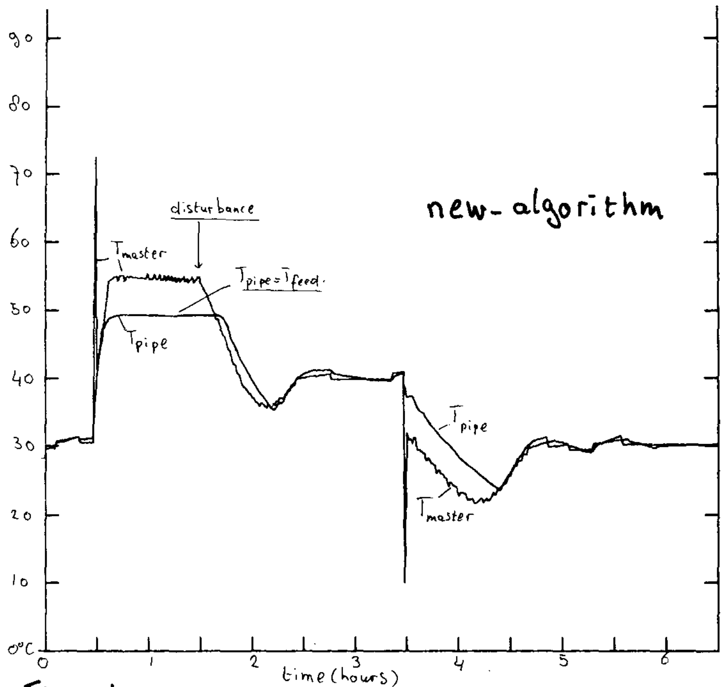

Fig. 4b.

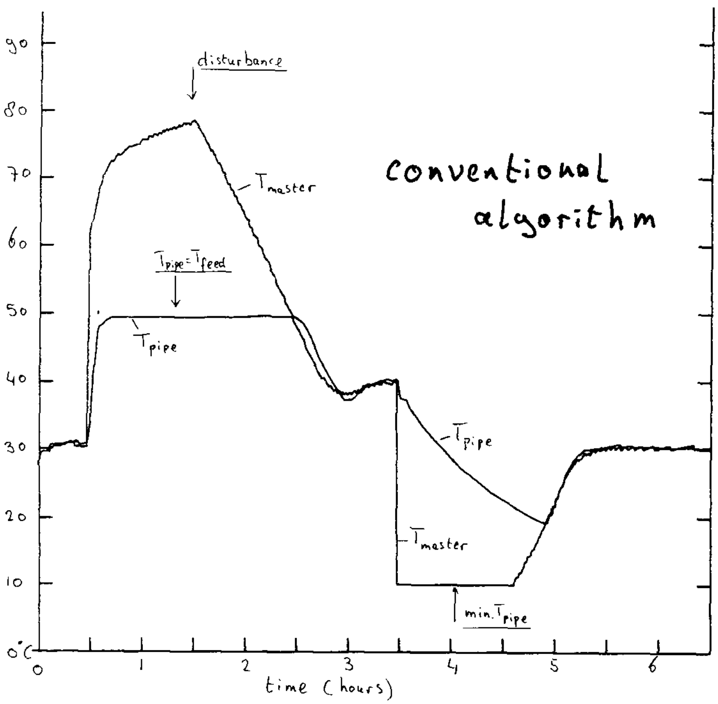

Fig. 4c. Comparison of control algorithms in simulation.