Smith (1975) concerning 0.54 mm β-naphtol granular particles and with the data of Schwartz et al. (1976) concerning 0.6 mm alumina particle has shown that Equation (4) give values of holdup which are at least 30% lower.

ACKNOWLEDGMENT

The authors wish to acknowledge The Institut Français du Petrole for the partial financial support of this study.

NOTATION

A = coefficient in Equation (1)
a = exponent in Equation (1)
d = particle diameter (m)
G = superficial gas flow rate (kg/m^2 · s)
Ga = Galileo number = $(d^3 g \rho_L{}^2/\mu_L{}^2)$
h_K, h_B = constant in an Ergun type of equation (h_K = 4.796; h_B = 0.245)
L = superficial liquid flow rate (kg/m^2 · s)
Re = Reynolds number = $(L\ d/\mu_L)$
Z = reactor length (m)

Greek Letters

β, β_{nc}, β_c = total, noncapillary and capillary holdup expressed in percentage of interparticle void $\beta = \beta_{nc} + \beta_c$
ϵ = porosity (interparticle)
λ = flow parameter $= \left[\dfrac{\rho_G}{\rho_{\text{air}}} \cdot \dfrac{\rho_L}{\rho_{\text{wat}}}\right]^{1/2}$
μ_L, μ_{wat} = liquid and water viscosity (cp)
ρ_G, ρ_L, ρ_{wat} = gas, liquid and water density (g/cm^3)
σ_L, σ_{wat} = liquid and water surface tension (dyne/cm)
ψ = flow parameter $= \dfrac{\sigma_{\text{wat}}}{\sigma_L} \cdot \left[\dfrac{\mu_L}{\mu_{\text{wat}}}\left(\dfrac{\rho_{\text{wat}}}{\rho_L}\right)^2\right]^{1/3}$
χ, χ' = gas-liquid flow parameters defined by Charpentier and Favier (1975)

LITERATURE CITED

Charpentier, J. C., C. Prost, W. P. M. Van Swaaij, and P. Le Goff, "Etude de la rétention de liquide dans une colonne à garnissage arrosé à co-courant et à contre-courant de gaz-liquide," *Chimie et Industrie, Génie Chimique*, **99**, 803 (1968).

Charpentier, J. C., and M. Favier, "Some Liquid Holdup Experimental Data in Trickle-Bed Reactors for Foaming and Nonfoaming Hydrocarbons," *AIChE J.*, **21**, 1213 (1975).

Charpentier, J. C., "Recent Progress in Two-Phase Gas-Liquid Mass Transfer in Packed Beds," *Chem. Eng. J.*, **1**, 161 (1976).

Colombo, A. J., G. Baldi, and S. Sicardi, "Solid-Liquid Contacting Effectiveness in Trickle-Bed Reactors," *Chem. Eng. Sci.*, **3**, 1101 (1976).

Goto, S., and J. M. Smith, "Trickle-Bed Reactor Performance, Part I: Holdup and Mass Transfer Effects," *AIChE J.*, **21**, 706 (1975).

Henry, H. C., and J. B. Gilbert, "Scale Up of Pilot Plant Data for Catalytic Hydroprocessing," *Ind. Eng. Chem. Process Design Develop.*, **12**, 328 (1973).

Larkins, R. P., R. R. White, and D. W. Jeffrey, "Two-Phase Concurrent Flow in Packed Beds," *AIChE J.*, **7**, 231 (1961).

Mears, D. E., "The Role of Liquid Holdup and Effective Wetting in the Performance of Trickle Bed Reactors," *Chem. Reac. Eng.*, **2**, *Adv. in Chem. Ser., No. 133*, 218 (1974).

Midoux, N., M. Favier, and J. C. Charpentier, "Flow Pattern, Pressure Loss and Liquid Holdup Data in Gas-Liquid Downflow Packed Beds with Foaming and Nonfoaming Hydrocarbons," *J. Chem. Eng. Japan*, **9**, 350 (1976).

Paraskos, J. A., J. A. Frayer, and Y. T. Shah, "Effect of Holdup in Complete Catalyst Wetting and Back Mixing During Hydroprocessing in Trickle Bed Reactors," *Ind. Chem. Process Design Develop.*, **14**, 315 (1975).

Sato, Y., T. Hirose, F. Takahashi, and M. Toda, "Pressure Loss and Liquid Holdup in Packed Bed Reactor with Co-current Gas-Liquid Downflow," *J. Chem. Eng. Japan*, **6**, 147 (1973).

Satterfield, C. N., A. A. Pelossof, and T. K. Sherwood, "Mass Transfer Limitations in a Trickle Bed Reactor," *AIChE J.*, **15**, 226 (1969).

Satterfield, C. N., and P. F. Way, "The Role of the Liquid Phase in the Performance of a Trickle Bed Reactor," *AIChE J.*, **18**, 305 (1972).

Satterfield, C. N., "Trickle Bed Reactors," *ibid.*, **21**, 209 (1975).

Schwartz, J. G., E. Weger, and M. P. Dudukovic, "Liquid Holdup and Dispersion in Trickle Bed Reactors," *ibid.*, **22**, 953 (1976).

Specchia, V., and G. Baldi, "Pressure Drop and Liquid Holdup for Two-Phase Concurrent Flow in Packed Beds," *Chem. Eng. Sci.*, **32**, 515 (1977).

Manuscript received June 24, 1977; revision received September 27, and accepted October 12, 1977.

A General Direct Digital Control Algorithm for a Class of Linear Systems

RAJEEV GAUTAM

and

RAJAKKANNU MUTHARASAN

Department of Chemical Enoineering
Drexel University
Philadelphia, Pennsylvania

Owing to a large increase in the use of digital computers for process control purposes, it has become increasingly important to develop practical direct digital control algorithms. Several special purpose algorithms, both continuous and discrete, have been reported in the literature for lumped parameter systems. In an early paper, Mosler et al. (1967) reported minimal prototype algorithms for a first-order system with dead time. Shunta and Luyben (1971) reported minimal prototype and minimum squared error design for a process with inverse response behavior. Luyben (1972) presented damping coefficient design charts for sampled data control of first-order processes with dead time. Although response of these special purpose algorithms are excellent for the specific tasks they are designed for, their performance often deteriorates under undesigned load condition or parameter shifts. In this paper we present a direct digital control algorithm for a first-order process with time delay, whose performance is as good as minimal prototype algorithms and has the advantage of responding better to parameter shifts. Furthermore, the algorithm responds to both set point change and load equally well.

THE CONTROL SYSTEM

In this paper we will restrict our analysis to the class of processes whose dynamics can be described by the following transfer function:

$$G_p(s) = K_p \frac{e^{-a\tau s}}{\tau s + 1} \qquad (1)$$

Several investigators have shown that the above model represents a large class of chemical process systems (see Ziegler and Nicolas, 1942; Mosler et al. 1967).

Figure 1 shows a block diagram of the control system. The output of the process is sampled every sampling period, and the manipulated variable value is calculated using an algorithm. The updated value of the manipulated variable is put out to the final control element through a holding circuit. Few of the past values of the process output and manipulated variable values together with the process parameters are stored in the process control computer.

DEVELOPMENT OF THE ALGORITHM ($a\tau = 0$)

The algorithm is designed by considering the input to the process as a series of step changes occurring at sampling instants. For simplicity, let the delay time be zero ($a = 0$). A general input to the process is shown in Figure 2. Let x_{i-1} denote the input to the process between the time $(i - 1)T$, and iT, where T is the sampling period. The output of the system, y at time iT, denoted by y_i is

$$y_i = y_{i-1} + \left(x_{i-1} - \frac{y_{i-1}}{K_p} \right) K_p(1 - e^{-T/\tau}) \qquad (2)$$

Note that we have approximated the input to the process to be constant at the value of x_{i-1} between the time $(i - 1)T$ and iT. From Equation (2) we may write

$$x_{i-1} = \frac{1}{\alpha} y_i + \left(\frac{1}{K_p} - \frac{1}{\alpha} \right) y_{i-1} \qquad (3)$$

where α is $K_p[1 - \exp(-T/\tau)]$. Since the manipulated variable $m(t)$ is kept constant between the time $(i - 1)T$ and iT at m_{i-1}, the load variable value during the same interval of time can be calculated as

$$L_{i-1} = \frac{1}{\alpha} y_i + \left(\frac{1}{K_p} - \frac{1}{\alpha} \right) y_{i-1} - m_{i-1} \qquad (4)$$

Furthermore, for the process output to be at the set point value, we require the input to the process x to be at $y^{(\text{set})}/K_p$. Equation (4) gives the equivalent load condition that has occurred, and we require the controller to compensate the difference. Hence, the manipulated variable for the following period [between iT and $(i + 1)T$] should be adjusted to

$$m_i = \frac{y^{(\text{set})}}{K_p} - \left\{ \frac{1}{\alpha} y_i + \left(\frac{1}{K_p} - \frac{1}{\alpha} \right) y_{i-1} - m_{i-1} \right\} \qquad (5)$$

The above equation enables the computer to calculate the manipulated variable at the time iT, using the present and the past value of the process output, and the manipulated variable. The above algorithm is in a sense a steady state feedforward algorithm in a feedback loop, the difference being that it estimates the load conditions instead of directly sensing it. Also, deviations in the process output from set point value due to deviations in process parameters or set point changes will be considered as an equivalent inlet load condition, and the algorithm will compensate accordingly. Such a feature enables the algorithm to eliminate any steady state error. The speed of recovery will depend on the time constant of the process. This might be undesirable when the process time constant is large. Hence, an additional compensation, proportional to the error, will greatly increase the speed of response. That is, the algorithm takes on the following form:

$$m_i = \frac{y^{(\text{set})}}{K_p} - \left\{ \frac{1}{\alpha} y_i + \left(\frac{1}{K_p} - \frac{1}{\alpha} \right) y_{i-1} - m_{i-1} \right\} + K\,[y^{(\text{set})} - y_i] \qquad (6)$$

Here K is a constant to be chosen either through simulations or on stream. Note that as the process output approaches the set point value, the compensation contributed by the proportional term tends to zero. Thus, near the steady state, the compensation is mainly done by the algorithm given by Equation (5).

CONTROL ALGORITHM WHEN TIME DELAY IS PRESENT

Let the delay time $a_\tau = \lambda T$. The derivation presented in this section can easily be extended to the case when the

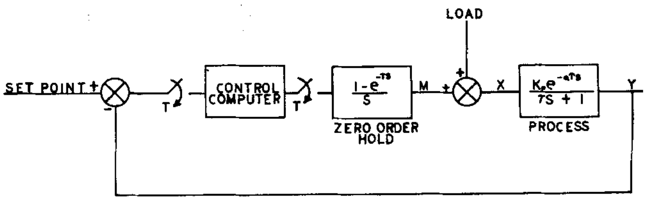

Fig. 1. Block diagram of the control system.

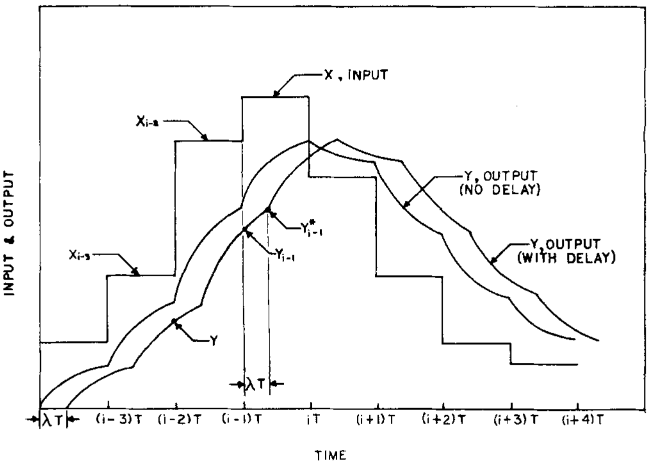

Fig. 2. Input-output relationships.

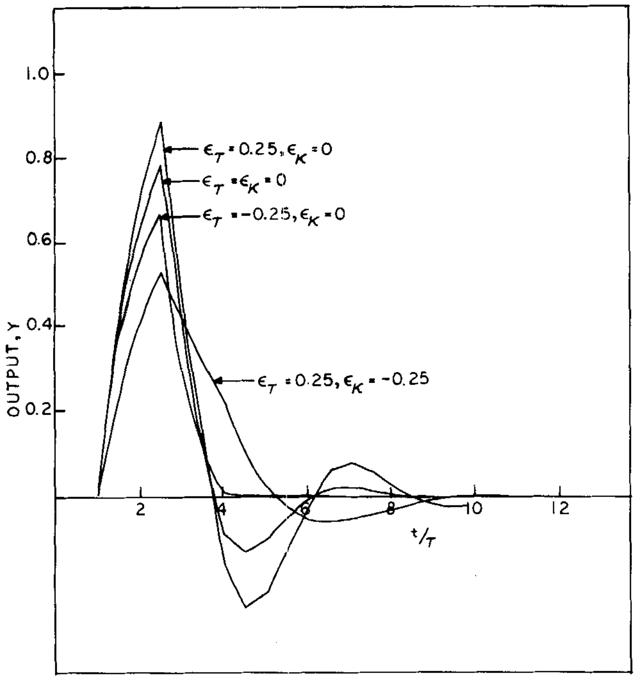

Fig. 3. Response of the proposed algorithm to step change in load. $T = 0.5, K_p = 1, \tau = 1, a = 1, K = 0.5$.

delay term is given by $a_T = (n + \lambda)T$, where n is an integer. A typical input-output relationship is shown in Figure 2. From the dynamics of the system

$$y_i = y^*_{i-1} + \left(x_{i-1} - \frac{y^*_{i-1}}{K_p}\right) K_p[1 - e^{-(1-\lambda)T/\tau}] \tag{7}$$

where

$$y^*_{i-1} = y_{i-1} + \left(x_{i-2} - \frac{y_{i-1}}{K_p}\right) K_p(1 - e^{-\lambda T/\tau}) \tag{8}$$

where the starred values of y_i are estimated values of the output of the system at the time $iT + \lambda T$. From the above equations, the input to the process and hence an estimate of the load condition can be calculated, and the algorithm of the form of Equation (5) becomes

$$m_i = \frac{y^{(\text{set})}}{K_p} - \left\{\frac{1}{\beta} y_i + \left(\frac{1}{K_p} - \frac{1}{\beta}\right) y^*_{i-1} - m_{i-1}\right\} \tag{9}$$

where β is defined by

$$\beta = K_p\{1 - \exp[-(1 - \lambda)T/\tau]\} \tag{10}$$

The above algorithm can be modified for speed of response as before by adding proportional action. One can easily show that when $a_T = nT + \lambda T$, the algorithm will take the form

$$m_i = \frac{y^{(\text{set})}}{K_p} - \left\{\frac{y_i}{\beta} + y^*_{i-1}\left(\frac{1}{K_p} - \frac{1}{\beta}\right) - m_{i-(n+1)}\right\} \tag{11a}$$

where

$$y^*_{i-1} = y_{i-1} + \left[x_{i-(n+2)} - \frac{y_{i-1}}{K_p}\right] K_p(1 - e^{-\lambda T/\tau}) \tag{11b}$$

Note that the action of the control algorithm is to compensate for the estimated load condition that existed n sampling instants ago. One should also recognize that since the load conditions during various sampling periods are estimated, the structure of the present algorithm lends itself to easy extensions to schemes incorporating prediction features. In the present work, we will restrict our attention to the algorithms of the form given by Equation (9).

SIMULATION AND DISCUSSION

The response of the process with the proposed control algorithm (with the proportional constant set at 0.5) to step changes in load and set point are presented in Figures 3 and 4. In the same figures, the responses when there is an error in the knowledge of the process parameters are also presented. For simulation purposes, the deviations were included by

$$\tau_p = \tau_a(1 + \epsilon_\tau) \tag{12}$$

$$K_{pp} = K_{pa}(1 + \epsilon_K) \tag{13}$$

The responses of the proposed algorithm are good even when $\epsilon_\tau = 0.25$ and $\epsilon_K = -0.25$. Typical responses are presented in Figures 3 and 4 when the errors in process parameters are ±25%.

For comparison purposes, the minimal algorithms of Mosler et al. (1967) were simulated with the errors in the process parameters and are presented in Figures 5 and 6. Although the response of the set point minimal algorithm is stable when errors of ±25% exist in process parameters, the response of the load minimal becomes unstable when $\epsilon_\tau = 0.25$ and $\epsilon_K = -0.25$. As a check, the characteristic equation of the control system was examined, and it was found that one of the roots lies outside the unit circle at $-0.63 \pm 0.85i$ which is at a radius of 1.058. In fact, when ϵ_τ is negative and ϵ_K is positive, the roots leave the unit circle even for small errors in the parameters. For example, when $\epsilon_\tau = -0.15$ and $\epsilon_K = +0.15$, one of the roots is at -1.045. Errors of magnitude 15% can easily enter in the determination of process parameters either as a modeling error (such as linearization) or owing to the shifting of the process itself.

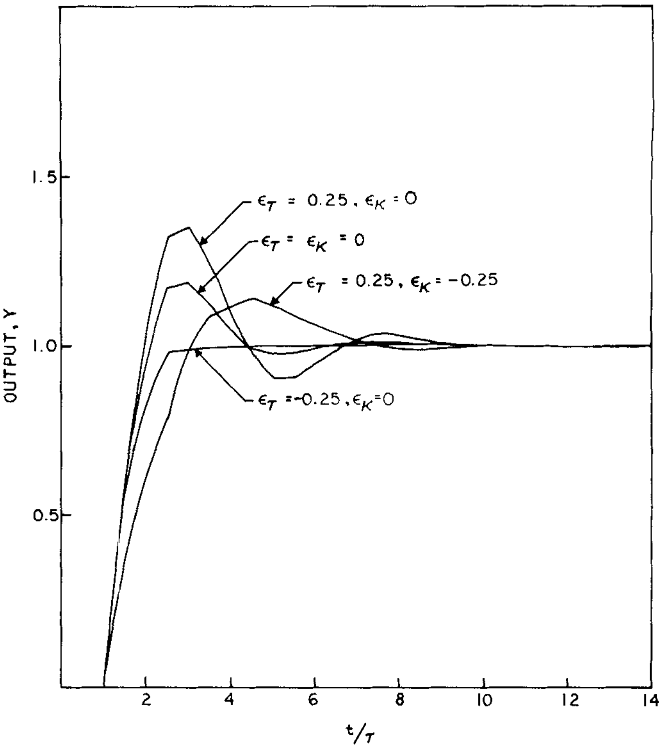

Fig. 4. Response of the proposed algorithm to step change in set point. $T = 0.5, K_p = 1, \tau = 1, a = 1, K = 0.5$.

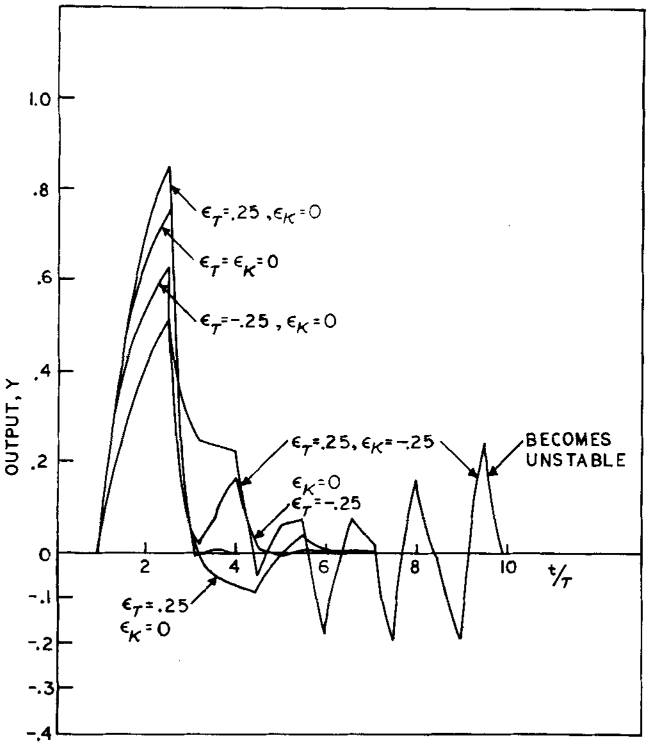

Fig. 5. Response to the minimal prototype load algorithm to step change in load. $T = 0.5$, $K_p = 1$, $\tau = 1$, $a = 1$.

In order to compare with conventional discrete proportional-integral algorithm, simulations were carried out for different numerical values of the parameters K_c and τ_I, the algorithm being

$$m_i = m_{i-1} + K_c \left(e_i - e_{i-1} + \frac{T}{\tau_I} e_i \right) \qquad (14)$$

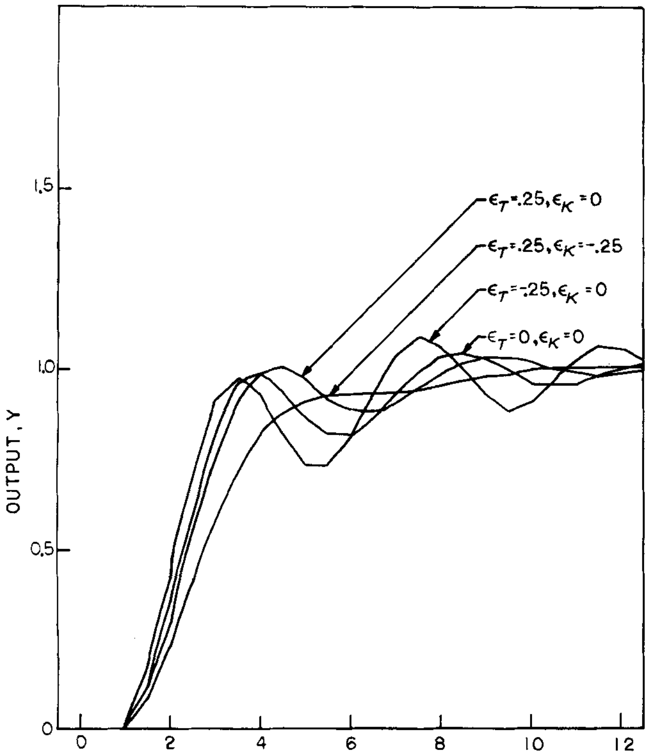

Fig. 7. Response of discrete proportional-integral control algorithm to step change in set point. $T = 0.5$, $K_p = 1$. $\tau = 1$, $a = 1$, $K_c = 1.0$, $\tau_I = 1.5$.

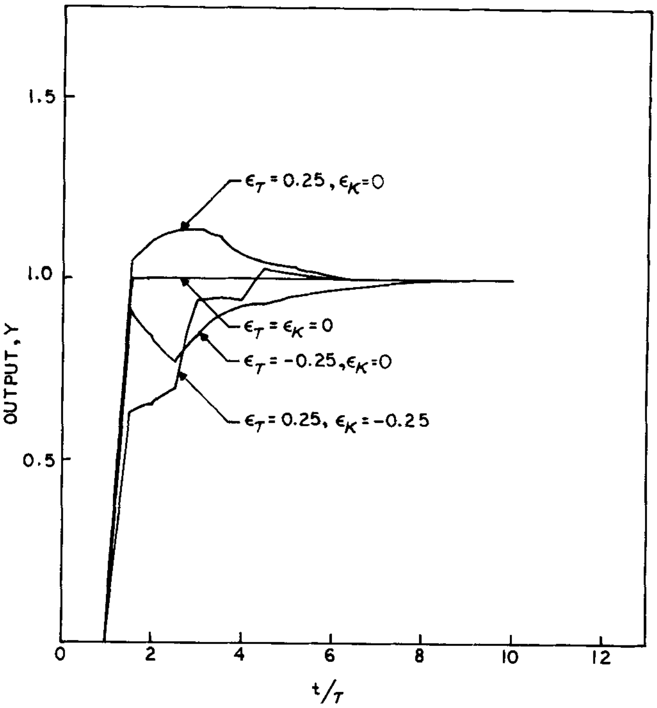

Fig. 6. Response of the minimal prototype set point algorithm to step change in set point. $T = 0.5$, $K_p = 1$, $\tau = 1$, $a = 1$.

For $K_p = a = \tau = 1$, the optimal choice was $K_c = 1$ and $\tau_I = 1.5$ based on integral of absolute error, decay ratio, and settling time Sample simulations are presented in Figures 7 and 8. As shown in Figure 8, the response of the process to step change in load becomes very oscillatory when $\epsilon_T = -0.25$ and $\epsilon_K = 0$. In general, the discrete P.I. algorithm has larger settling time and larger decay ratio when compared with the algorithm proposed in this note. Furthermore, it was found that if the P.I. algorithm

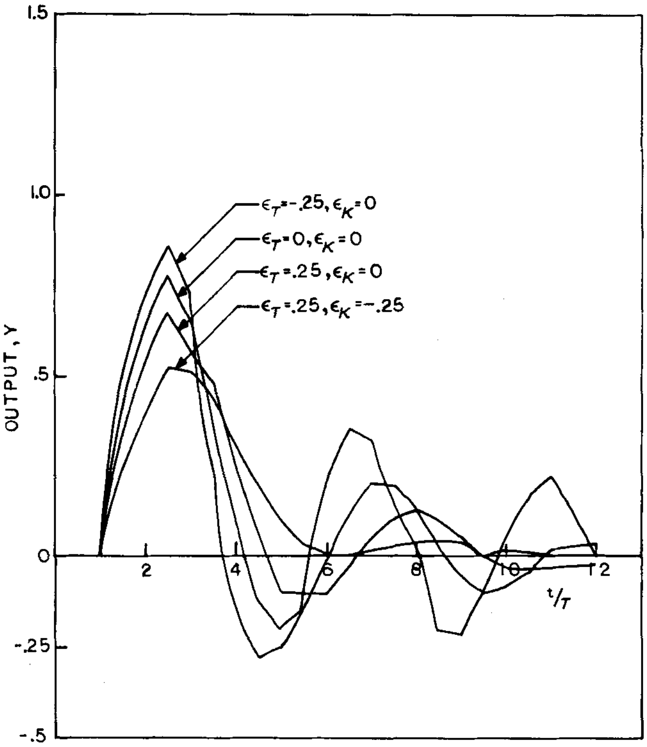

Fig. 8. Response of discrete proportional-integral control algorithm to step change in load. $T = 0.5$, $K_p = 1$, $\tau = 1$, $a = 1$, $K_c = 1.0$, $\tau_I = 1.5$.

is tuned for set point response, the same setting gives very poor load response, and vice versa.

The response of the proposed algorithm does not deteriorate when errors are present in the process parameters. Furthermore, only one algorithm is needed to compensate for both set point and load changes. This is a distinct advantage over the previously reported minimal algorithms and discrete P.I. algorithms. Since the proposed algorithm views deviations in the output of the process from the set point value as an equivalent step disturbance in the input, the response is always stable.

ACKNOWLEDGMENT

The authors thank Louis Haas who assisted in the preparations of the drawings.

NOTATION

a = constant used to define dead time
$G_p(s)$ = process transfer function
K_p, K_c = process gain, controller gain
m = manipulated variable
T = sampling period
X' = input to the process
y = output of the process

Greek Letters

α = constant defined in Equation (2)
β = constant defined by Equation (9)
ϵ_τ = deviation in process parameter, τ
ϵ_k = deviation in process parameter, K_p
λ = fraction used to define time delay
τ, τ_I = process time constant, integral time

Subscripts

a = algorithm
p = process
i = value of the variable between the time iT and $(i+1)T$

Superscripts

set = set point of the variable
$*$ = variable evaluated λT time later

LITERATURE CITED

Luyben, W. L., "Damping Coefficient Design Charts for Sampled-Data Control of Processes with Deadtime," *AIChE J.*, **18**, No. 5, 1048 (1972).

Mosler, H. A., L. B. Koppel, and D. R. Coughanowr, "Process Control by Digital Compensation," *ibid.*, 768 (1967).

Shunta, J. P., and W. L. Luyben, "Studies of Sampled-Data Control of Distillation Columns. Feedback Control of Bottoms Composition with Inverse Response Behavior," *Ind. Eng. Chem. Fundamentals*, **10**, No. 3, 486 (1971).

Ziegler, J. C., and N. B. Nichols, "Optimum Settings for Automatic Controllers," *Trans. Am. Soc. Mech. Engrs.*, **64**, 759 (1942).

Manuscript received November 18, 1976; revision received August 8, and accepted October 12, 1977.

Reaction Near the Grid in Fluidized Beds

JOHN R. GRACE
and
HUGO I. DE LASA

Department of Chemical Engineering
McGill University
Montreal, Canada

There is growing recognition that two-phase models for fluidized-bed reactors should take account of end effects. Experimental results show that bubbling-bed models do not properly represent reaction near the gas distributor (Chavarie and Grace, 1975) or in the freeboard region (Furusaki et al., 1976). In this note we consider reaction in the vicinity of the grid.

EARLIER MODEL

Behie and Kehoe (1973) proposed a two-phase model which extends one of the well-known Orcutt models (Orcutt et al., 1962) by treating a plug flow, jet region devoid of particles in series with the bubble phase. Gas in the entire dense phase is assumed to be perfectly mixed. Since the resistance to mass transfer between grid jets and the dense phase is much lower (Behie, 1972) than that between bubbles and the dense phase, the grid model predicts significantly higher conversions than the corresponding Orcutt model, providing that the kinetic rate constant is sufficiently large for the rate of reaction to be controlled by hydrodynamic factors.

The case considered by Behie and Kehoe involves a first-order, constant volume, gas phase reaction $A \rightarrow B$ catalyzed by the solid particles. Temperature gradients are neglected. The flow patterns assumed for the three phases are shown schematically in Figure 1. Note that the entire flow of gas UA is assumed to pass through the jets and then to divide itself between the bubble phase and dense phase at the top of the jets. The mass balance equations for the three phases are

Jet phase $(0 \leq x \leq h)$:

$$U \frac{dC_j}{dx} + k_j a_j (C_j - C_d) = 0 \qquad (1)$$

Bubble phase $(h < x \leq H)$:

$$\beta U \frac{dC_b}{dx} + k_b a_b (C_b - C_d) = 0 \qquad (2)$$

Dense phase $(0 \leq x \leq H)$:

$$U(1-\beta)(C_d - C_{jh}) + \int_0^h k_j a_j (C_d - C_j)\,dx + \int_h^H k_b a_b (C_d - C_b)\,dx + k_r C_d H_{mf} = 0 \qquad (3)$$

0001-1541-78-9913-0364-$00.75 © The American Institute of Chemical Engineers, 1978.