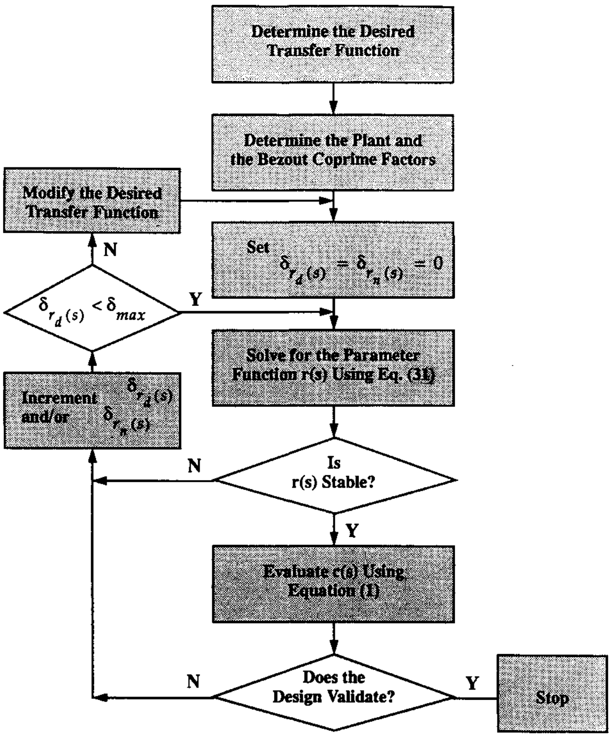

Fig. 1 Design synthesis procedure.

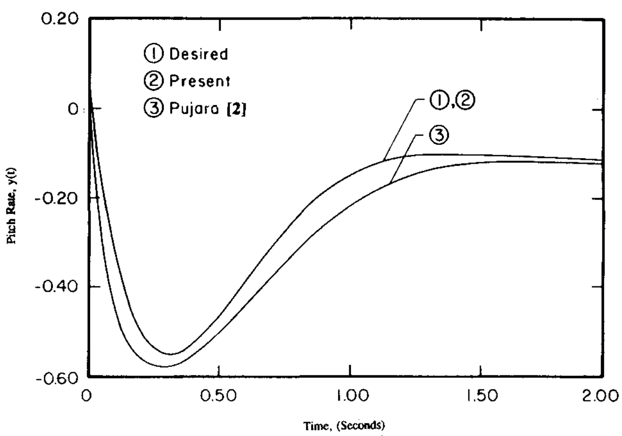

Fig. 2 Unit-step response comparisons for demonstration example problem.

Using Eq. (1), the corresponding compensator will then be given by the following expression.

$$C(s) = \frac{0.18677 + 0.0172919s + 0.0154213s^2}{1.1197 + 0.62381s + 0.06240s^2} \tag{29}$$

The unit step responses are depicted in Fig. 2, and it is apparent that our design meets the objectives more closely than that previously obtained.

Summary and Conclusions

The solution to the closed-loop model-matching problem based on a factorization approach requires one to search the set of all admissible function parameters. A possible search procedure was presented, and the solution space was given in terms of a set of linear algebraic equations, i.e., Eq. (25). The procedure is computer-assisted, and it is based on minimizing a mean-square error criteria.

The procedure may be extended to account for minimizing the sensitivity function as well as to account for integrity conditions. In both cases, the solution space will remain linear.

Acknowledgment

This research was in part funded by the Department of Defense.

References

[1]Nassirharand, A., and Patwardhan, A., "A Fast Algorithm for Obtaining Low-Order Coprime Fractional Representations of SISO Systems," *Control Theory and Advanced Technology,* Vol. 4, No. 2, 1988, pp. 177–281.

[2]Pujara, L. R., "Computer-Aided Control Systems Design Technique with Applications to Aircraft Flying Qualities," *Journal of Guidance, Control, and Dynamics,* Vol. 11, No. 5, 1988, pp. 250–255.

[3]Chen, C. F., and Shieh, L. S., "An Algebraic Method for Control System Design," *International Journal of Control,* Vol. 11, No. 5, 1970, pp. 717–739.

[4]Bongiorno, J. J., and Youla, D. C., "On the Design of Single-Loop Single-Input-Output Feedback Control Systems in the Complex Frequency Domain," *IEEE Transactions on Automatic Control,* Vol. AC-22, 1977, pp. 416–423.

[5]Francis, B. A., *A Course in H_∞ Control Theory,* Springer-Verlag, New York, 1988.

[6]Youla, D. C., Bongiorno, J. J., and Jabr, H. A., "Modern Wiener-Hopf Design of Optimal Controllers—Pt. 1: The Single-Input Case," *IEEE Transactions on Automatic Control,* Vol. AC-21, 1976, pp. 3–14.

[7]Doyle, J. C., Glover, K., Khargonekar, P. P., and Francis, B. A., "State-Space Solutions to Standard H_2/H_∞ Control Problems," *IEEE Transactions on Automatic Control,* Vol. AC-34, No. 8, 1989, pp. 831–847.

[8]Zhou, K., Doyle, J., Glover, K., and Bodenheimer, B., "Mixed H_2 and H_∞ Control," *Proceedings of the 1990 American Control Conference* (San Diego CA), Inst. of Electrical and Electronics Engineers, Piscataway, NJ, pp. 2502–2507.

[9]Nassirharand, A., and Patwardhan, A., "Parametric Synthesis of Single-Loop Automatic Control Systems," *Computer-Aided Design,* Vol. 22, No. 5, 1990, pp. 301–308.

Parametric Study of Adaptive Generalized Predictive Controllers

William L. Brogan*
University of Nevada, Las Vegas, Nevada 89154
and
Sung-Duck Han†
Daewoo Engineering Company, Seoul, Korea

Introduction

OPTIMAL tracking problems with finite and moving horizons have been studied extensively by Kishi.[1] Brickner and Brogan[2] proposed a controller model that included adaptive model estimation and predictive control to adjust nominal control inputs. The combination of the model identification, predictive controller, and the finite control horizon is essential to what is now called the generalized predictive controller extensively reported by Clarke.[3-9] This Note presents results of a parametric study of generalized predictive control (GPC), based largely on the thesis of Han.[10]

Generalized Predictive Control

The GPC controller minimizes a quadratic cost function over a receding horizon to calculate future control inputs. The system is modeled as

$$A(z^{-1})y(t) = B(z^{-1})u(t-1) + e(t) \tag{1}$$

Received May 6, 1991; revision received Nov. 24, 1991; accepted for publication Jan. 13, 1992.

*Professor and Chair, Department of Electrical and Computer Engineering.

†Senior Engineer.

Any dead-time or pure time delay k is absorbed into the polynomial $B(z^{-1})$ so that the leading $k-1$ elements are zero. The unknown model parameters, i.e., the coefficients in the A and B polynomials, are defined as the vector θ, which is estimated using recursive least squares:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)[y(t) - X(t)\hat{\theta}(t-1)]$$
$$P'(t-1) = P(t-1)/\beta$$
$$K(t) = P'(t-1)X^T(t)/[1 + X(t)P'(t-1)X^T(t)] \qquad (2)$$
$$P(t) = [1 - K(t)X(t)]P'(t-1)$$

$X(t)$ is the row vector of previous outputs and inputs that multiply the unknown coefficients θ in Eq. (1). The leading coefficient of $A(z^{-1})$ is normalized to unity. The forgetting factor is β, and P is (essentially) the covariance matrix of the parameter estimation errors.

The predictive control law functions as follows. The current process output $y(t)$ is used to predict $p(t+j)$, the future output j time steps ahead, using estimated model parameters $\hat{\theta}(t)$:

$$p(t+j) = G_j\Delta u(t+j-1) + F_j y(t) \qquad (3)$$

The polynomials G_j and F_j can be computed by solving the Diophantine equation

$$1 = E_j(z^{-1})A\Delta + z^{-j}F_j(z^{-1}) \qquad (4)$$

where $\Delta = 1 - z^{-1}$. Then $G_j(z^{-1}) = E_jB(z^{-1})$ is used. Clarke[3] give an efficient recursive procedure for finding F_j and G_j for all j up to some planning horizon N_2. Then an $N \times N_u$ matrix G can be formed from the coefficients g_α of $z^{-\alpha}$ in the polynomial G:

$$G = \begin{bmatrix} g_0 & 0 & 0 \\ g_1 & g_0 & 0 \\ & & \\ g_{N-1} & g_{N-2} & g_{N-N_u} \end{bmatrix} \qquad (5)$$

Then the vector of future outputs

$$Y = [y(t+1) \; y(t+2) \cdots y(t+N_2)]^T$$

can be written in terms of future incremental control changes

$$U = [\Delta u(t), \ldots, \Delta u(t+N_U-1)]^T$$

and predicted future outputs

$$\hat{Y} = [p(t+1) \; p(t+2) \cdots p(t+N_2)]^T$$

as

$$Y = GU + \hat{Y} + V \qquad (6)$$

where V is a zero mean random vector. Define the future error as $r(t+j) = w(t+j) - y(t+j)$. The set of future incremental control changes $\Delta u(t+j-1)$ is sought that minimizes

$$J(N_1, N_2, N_u, \lambda) = E\left\{ \sum_{j=N_1}^{N_2} r^2(t+j) + \lambda \sum_{j=1}^{N_u} \Delta u^2(t+j-1) \right\} \qquad (7)$$

Using the previously defined data vectors along with the vector W_d of N_2 known future set-point values

$$W_d = [w(t+1) \; w(t+2) \cdots w(t+N_2)]^T$$

gives the minimizing control sequence

$$U = [G^TG + \lambda I)^{-1}G^T[W_d - \hat{Y}] \qquad (8)$$

Table 1 Controller/estimator parameters

Model	Sampling period T
	Orders of A and B polynomials (known or unknown)
	Known or unknown values of a_j and b_j
	System characteristics (stable or minimum phase)
	Set-point function $w_d(t)$
Estimator	Forgetting factor β
	Initial parameter uncertainty, $P(0)$
	Initial estimates, $\hat{\theta}(0)$
	Startup period T_{start} and control $u_{\text{start}}(t)$
	Estimation update cycle rate, every rth control cycle
Controller	Minimum and maximum output horizons N_1 and N_2
	Control horizon N_u
	Cost function weighting factor λ
	Control limits $U_{\min}$, $U_{\max}$
	Control update cycle period T
Noise levels	Mean and variance of output and control disturbances

Although there are N_u future control increments in this vector, only the first one is used, giving the next control input

$$u(t) = u(t-1) + g^T(W_d - \hat{Y}) \qquad (9)$$

where g^T is the first row of $(G^TG + \lambda I)^{-1}G^T$. This process is repeated each control cycle, after shifting the previous output, control, and set-point data and adding one new element in each array.

Simulation Study

The many parameters and considerations in applying the GPC controller are summarized in Table 1. To gain insight into the impact of parameter selection, Han[10] studied nine categories of systems/problem types. These include systems with unknown models, driven by input and output noise, nonminimum phase models, models with unknown or time-varying delay, problems involving overparameterization and underparameterization, and studies of the effect of the cost weighting factor λ and of the sampling rate. Comparisons with fixed proportional-integral derivative (PID) controllers and moving horizon linear quadratic (LQ) controllers have been made. Control limits are implemented simply by clipping the computed control whenever it exceeds a limit.

Typical Results

Kishi[1] considered the system $H(s) = 0.5/[s(s+0.5)]$, sampled at $T = 1$ s intervals, giving the discrete model $y(t) = 1.6065y(t-1) - 0.6065y(t-2) + 0.21306u(t-1) + 0.1804u(t-2)$. The order of the model is assumed known, but not the coefficient values. Thus, there are four unknowns in the θ vector, two A and two B coefficients. With $N_u = N_1 = 1$, $\hat{\theta}(0) = [1\,0\,0\,0]^T$, $\lambda = 0$, $\beta = 1$, $P(0) = 100I$, $r = 3$, $T_{\text{start}} = 10$, $u_{\text{start}} = 0.1$, $U_{\max} = 5.5$, $U_{\min} = -5.5$, and no disturbances; Fig. 1 gives the performance for $N_2 = 2$, 4, and 8. Note that $N_2 = 2$ agrees with the system time constant, since $T = 1$ s. The ability to follow the triangular set-point function is clearly inadequate. Also note that $N_2 = 4$ is better than $N_2 = 8$. This shows that use of too long a planning horizon merely slows down system response. When the same system is sampled at a more appropriate rate of $T = 0.25$ s (i.e., eight samples per time constant), the results of Fig. 2 are obtained. Now $N_2 = 4$ cycles (i.e., 1 s) comes the closest to the set point near cycle 180, but it is also the most oscillatory between 50–100 cycles. $N_2 = 8$ (2 s = one time constant) is somewhat smoother initially and almost as close to w_d later. These results suggest that selecting N_2 to correspond to the system response time is a good choice provided that the sample rate is adequate. The selections of T and N_2 are coupled together. Figure 3 presents typical results obtained from the underdamped system $H(s) = 1/[s^2 + 0.7s + 1]$ that has a rise time of about 2 s, a settling time of about 10 s, and about 29% overshoot to a step input. The peak occurs at approximately 3 s. A sample period $T = 1$ is used, giving $y(t) = 0.8349y(t-1) - 0.4966y(t-2) + 0.3704u$

$(t - 1 - k) + 0.29123u(t - 2 - k)$ where a delay of k cycles has been inserted. Clarke's[3] suggested default values have been used for $N_1 = N_u = 1$, $N_2 = 10$. Additional parameters were $U_{max} = -U_{min} = 40$, $T_{start} = 10$, $u_{start} = 10$, $r = 5$, $P(0) = 100I$, $\beta = 1$, and $\lambda = 0$. The model was overparameterized by assuming θ contained three unknown A coefficients and eight unknown B coefficients. A small amount of noise (0 mean, 0.1 standard deviation) was added to both y and u, starting at $T_\xi = 120$.

Typical results are shown in Fig. 3 for three different values of the delay k. The parameter estimates are not shown, but they did not always converge to their correct values. In spite of this, acceptable performance is achieved over a range of delays, using fixed default values for most parameters. Two general types of problems can (and occasionally did) arise, even with good choices of sample rates and controller horizons. One relates to the estimator algorithm experiencing numeric underflow. This happened in the results of Fig. 3 for $k \geq 5$ and was then avoided by using $\beta = 0.965$ and $P(0) = 1000I$. The other problem related to occasional numerical overflow. One cause of this is the matrix G^TG being near singular. This can be avoided by using a nonzero value of the

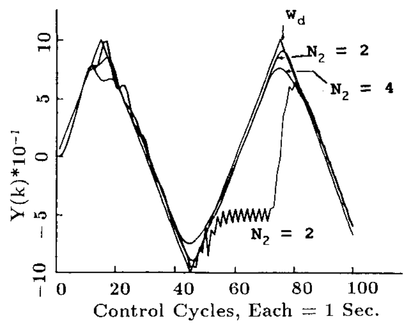

Fig. 1 Kishi's example with GPC controller turned on at k = 10 s.

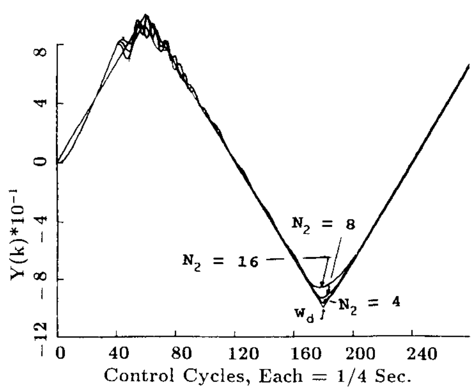

Fig. 2 Kishi's example with faster sample rate, T = 0.25 s.

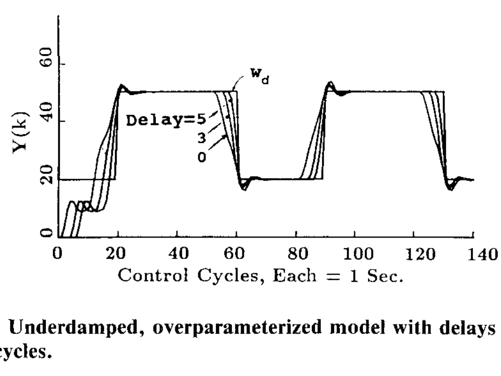

Fig. 3 Underdamped, overparameterized model with delays of 0, 3, and 5 cycles.

control weight λ. The other overflow problem was one of controller windup, where the control is at a saturation limit U_{max} or U_{min} and the commanded Δu continues to increase. This effect is exacerbated by noise when the desired control signal is already at or near its bounds. It seems that GPC can cope with the control signal saturating occasionally for brief periods. However, if the available limits are too small, difficulties can naturally be expected.

Conclusions

The GPC algorithm is an efficient and effective approach to the control of many (partially) unknown or slowly varying systems. Comparisons with fixed PID controllers, not reported here, show the superiority. Performance with unknown model parameters is essentially equivalent to that obtained by other methods with a known model. Optimal design parameters can be selected for a known model. When the model is largely unknown, the adequacy of the following parameter selection guidelines has been verified. A one-step ahead control horizon was adequate in all cases considered. The output horizon should be sufficiently long to cover the system response time, including pure delays. In all cases the output horizon should contain a minimum of four future cycles. Very crude initiation of the estimation process is adequate if a carefully controlled estimation startup is possible before closing the GPC loop. The number of startup estimation cycles should exceed the number of unknown model parameters being estimated. After this initial period, parameter updating may be slowed to perhaps every third to fifth control cycle if parameters vary slowly. The forgetting factor should be less than unity, with the best value depending upon the rate of change of system parameters. A nonzero control weighting should be used in the cost function. The major difficulty that arose while applying GPC to an unknown system was caused by control saturation. The control weighting factor can be increased to reduce control command levels. A better solution is to provide adequate control limits whenever possible. Noise also degrades system performance. Output noise seems more serious than noise on the control inputs. Noise can lead to biased parameter estimates, as can nonpersistently exciting inputs. Good output performance was commonly obtained even when some parameters failed to converge to their true values.

References

[1]Kishi, F. H., "On Line Computer Control Techniques and Their Application to Re-entry Aerospace Vehicle Control," *Advances in Control Systems,* Vol. 1, edited by C. T. Leondes, Academic Press, New York, 1964, Chap. 6.

[2]Brickner, T. S., and Brogan, W. L., "Cortical and Cerebellar Control of Motor Functions, an Optimal Adaptive Learning Control System," *8th Rocky Mountain Bioengineering Symposium* (Fort Collins, CO), Instrument Society of America, Pittsburgh, PA, May 1971.

[3]Clarke, D. W., and Gawthrop, P. J., "Self-Tuning Controller," *Proceedings of IEE,* Vol. 122, No. 9, 1975, pp. 929–934.

[4]Clarke, D. W., and Gawthrop, P. J., "Self-Tuning Control," *Proceedings of IEE,* Vol. 126, No. 6, 1979, pp. 633–640.

[5]Clarke, D. W., and Gawthrop, P. J., "Implementation and Application of Microprocessor-Based Self-Tuners," *Automatica,* Vol. 17, No. 1, 1981, pp. 233–244.

[6]Clarke, D. W., "Self-Tuning Control of Nonminimum-Phase Systems," *Automatica,* Vol. 20, No. 5, 1984, pp. 501–517.

[7]Clarke, D. W., and Zhang, L., "Long-Range Predictive Control Using Weighting-Sequence Models," *Proceedings of IEE,* Vol. 134, Pt. D, No. 3, 1987, pp. 187–195.

[8]Clarke, D. W., Mohtadi, C., and Tuffs, P. S., "Generalized Predictive Control: Pt. 1, The Basic Algorithm; Pt. 2, Extensions and Interpretations," *Automatica,* Vol. 23, No. 2, 1987, pp. 137–160.

[9]Clarke, D. W., "Application of Generalized Predictive Control to Industrial Processes," *IEEE Control Systems Magazine,* April 1988, pp. 49–54.

[10]Han, S.-D., "Self-Tuning Control Simulation Study Based on the Generalized Predictive Control Method," M. S. Thesis, Dept. of Electrical Engineering, Univ. of Nebraska, Lincoln, NE, Dec. 1988.