SELF-TUNING OPTIMAL ROBUST INTEGRATING POLE-PLACEMENT CONTROL

P. LÖHNBERG AND E.H. VAN DE WAAL

Dept. of El. Eng. and Mechatronics Research Center Twente. University of Twente, Enschede, The Netherlands

Abstract. At each sample, a polytope of guaranteed bounds is estimated for the process parameters in the presence of a bounded disturbance, and an optimal robust pole-placer is designed. The control design minimizes the summed squared distances between the coefficient vector of the closed loop polynomials for the vertices of the process parameter polytope and that for a desired polynomial. Steady-state errors due to the parameter uncertainty are averaged out or eliminated by an integrator at the cost of a lower bandwidth of the controlled system. The benefits of the algorithms are illustrated by simulations for a first-order process.

Keywords. Adaptive control; convergence; feedforward control; least-squares approximations; optimal control; performance bounds; pole placement; process parameter estimation; robust control; self-tuning regulator

1. INTRODUCTION

Only recently, guaranteed estimators for process parameter bounds have been combined with on-line robust control design to form adaptive robust controllers (Kuntzevich and Lychak, 1992; Veres and Norton, 1991). These are not to be confused with robust adaptive controllers which make conventional adaptive controllers robust to structural uncertainty and disturbances and sometimes erroneously are called adaptive robust too.

Kuntzevich and Lychak (1992) use state-space models which are complicated for SISO processes. Veres and Norton (1991) use the freedom to position the feasible poles in an allowed pole region for dual on-line experiment design to minimize the feasible parameter region. This enables to track time-variations, but makes the method rather complicated and hence not easy to implement in real time.

The duality is less relevant for time-invariant processes, where the pole-placement freedom can be used to optimize the performance, with robustness check and correction if needed. Also the steady-state regulation error by process parameter uncertainty needs attention. This error can be reduced by minimizing its average over the vertices, or removed by introducing an integrator. This paper describes the resulting straightforward controllers and illustrates them for a simple process.

2. PROCESS AND INITIAL UNCERTAINTY

2.1. Continuous-Time Model

The method is illustrated for the first-order time-invariant transfer function

$$H(s) = \frac{K}{\tau s + 1}, \quad (1)$$

where the nominal values of the gain K and time constant τ are taken equal to those of the transfer function from the amplifier input to the angular velocity of the motor of an existing servomechanism described by Löhnberg and Van Amerongen (1989) as $K^0 = 0.03$; $\tau^0 = 0.04$ s. Gross initial uncertainty is described by the orthonormal initial uncertainty region

$$\begin{aligned} K_{\min} &\leq K \leq K_{\max}, & \tau_{\min} &\leq \tau \leq \tau_{\max}, \\ K_{\min} &= K^0 / 2, & \tau_{\min} &= \tau^0 / 2, \\ K_{\max} &= 2 K^0, & \tau_{\max} &= 2 \tau^0. \end{aligned} \quad (2)$$

2.2. Discrete-Time Model

In practice, identification and control occur by digital computer. Therefore the continuous-time transfer function is transformed to the pulse-transfer operator

$$H(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})},$$

$$B(q^{-1}) = b_1 q^{-1} + \dots + b_m q^{-m},$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_n q^{-n}, m \leq n, \quad (3)$$

where n is the process order. For the example $b_1 = b$, $a_1 = a$, $m = n = 1$, where for a sampling interval h the relation with the continuous-time parameters is according to

$$b = (1 - e^{-h/\tau})K; \quad a = -e^{-h/\tau}. \quad (4)$$

By transformation (4), the continuous-time nominal values K^0 and τ^0 translate into discrete-time nominal values $b^0 = 3.4 \cdot 10^{-3}$, $a^0 = -0.88$, and the initial uncertainty region (2) for K and τ yields a non-rectangular initial uncertainty region for b and a , which for simplicity will be approximated conservatively by the worst-case orthonormal set

$$\begin{aligned} b_{\min} &\leq b \leq b_{\max}, & a_{\min} &\leq a \leq a_{\max}, \\ b_{\min} &= 0.88 \cdot 10^{-3}, & a_{\min} &= -0.94, \\ b_{\max} &= 13 \cdot 10^{-3}, & a_{\max} &= -0.78, \end{aligned} \quad (5)$$

yielding k (4 for the example) vertices θ_i , $i = 1, \dots, k$, of the polytope for the process parameter vector $\theta = [a_1 \dots a_n \ b_1 \dots b_m]^T$ which bounds the uncertainty region $\mathcal{L}$ shown in Fig. 1.

Fig. 1. Initial uncertainty region with vertex numbers and with nominal parameters indicated by X

2.3. Disturbance Model

As practical processes are disturbed, the relation between control samples $u(k)$ and process output samples $y(k)$ according to (3) is expressed in the ARX model

$$A(q^{-1})y(k) = B(q^{-1})u(k) + e(k), \quad (6)$$

where $e(k)$ is an uncorrelated equation disturbance sequence which is uniformly distributed between bounds $\pm\delta$ chosen to yield a signal-to-noise ratio 10 for the arbitrarily chosen range ± 1 for the control or reference signal respectively.

3. ROBUST POLE-PLACEMENT DESIGN

3.1. Non-Integrating Controller Structure

To prevent state measurement or estimation, output feedback with feedforward control is used:

$$\begin{aligned} R(q^{-1})u(k) &= T(q^{-1})u_C(k) - S(q^{-1})y(k); \\ R(q^{-1}) &= 1 + r_1 q^{-1} + \dots + r_{n-1} q^{-n+1}, \\ S(q^{-1}) &= s_0 + s_1 q^{-1} + \dots + s_{n-1} q^{-n+1}, \\ T(q^{-1}) &= t_0 + t_1 q^{-1} + \dots + t_{n-1} q^{-n+1}, \end{aligned} \quad (7)$$

where $u_C(k)$ is the reference signal. The resulting closed-loop pulse-transfer operator with numerator and characteristic closed-loop polynomial are

$$\begin{aligned} H_C(q^{-1}) &= \frac{D(q^{-1})}{P(q^{-1})}, \\ D(q^{-1}) &= B(q^{-1})T(q^{-1}), \\ P(q^{-1}) &= 1 + p_1 q^{-1} + \dots + p_{2n-1} q^{-2n+1}, \end{aligned} \quad (8)$$

where $P(q^{-1})$ follows from the Diophantine equation

$$P(q^{-1}) = R(q^{-1})A(q^{-1}) + S(q^{-1})B(q^{-1}). \quad (9)$$

3.2. Optimal Robust Pole-Placement

The coefficients of $R(q^{-1})$ and $S(q^{-1})$ can be represented by controller feedback parameter vector $\chi = [1 \ r_1 \ \dots \ r_{n-1} \ s_0 \ \dots \ s_{n-1}]^T$, and the characteristic closed-loop polynomial $P(q^{-1})$ according to (8) by its characteristic coefficient vector $\mathbf{p} = [1 \ p_1 \ \dots \ p_{2n-1}]^T$. Then an arbitrary controller χ yields different values $\mathbf{p}_i$ of $\mathbf{p}$ for different values θ_i of θ . Hence a unique choice of χ requires a criterion. A reasonable and simple criterion is to minimize the summed Cartesian distance

$$J(\chi) = \sum_{i=1}^k \|\mathbf{p}_i - \mathbf{p}_D\|^2 \quad (10)$$

between the coefficient vector $\mathbf{p}_D$ of a desired polynomial $P_D(q^{-1})$ and the coefficients $\mathbf{p}_i$, $i = 1$,

..., k , of the closed-loop polynomials $P_i(q^{-1})$ obtained by controlling a process with parameters θ_i using feedback χ . For the first-order process, the desired polynomial was chosen as $P_D(q^{-1}) = 1 - 0.4q^{-1}$. To express p_i into χ , the Diophantine equation (9) is written in vector notation as

$$\mathbf{p} = \mathbf{M}(\theta)\chi, \quad (11)$$

with process Sylvester matrix

$$\mathbf{M}(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ a_1 & 1 & b_1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ a_n & a_1 & b_n & b_1 \\ 0 & a_n & 0 & b_n \end{bmatrix}. \quad (12)$$

By the use of vector notation (11), cost (10) can be written as

$$J(\chi) = \sum_{i=1}^k \|\mathbf{M}(\theta_i)\chi - \mathbf{p}_D\|^2, \quad (13)$$

which is minimized by the optimal robust pole-placement controller found by least-squares optimization

$$\chi_O = \left(\sum_{i=1}^k \mathbf{M}^T(\theta_i)\mathbf{M}(\theta_i) \right)^{-1} \left(\sum_{i=1}^k \mathbf{M}^T(\theta_i)\mathbf{p}_D(\theta_i) \right). \quad (14)$$

The feedforward polynomial $T(q^{-1})$ was chosen such that it minimizes the steady-state error over all vertices.

3.3. Correction to Place the Poles in the Allowed Region.

If the process uncertainty is large, the described controller may not ensure performance robustness. Checking this requires an allowed pole region which is described by a set ρ_A within a polytope of allowed polynomial coefficients $\mathbf{p}_A$. In the example with its real poles, the allowed pole region is equal to the allowed set for p_1 which was chosen as $-0.99 \leq p_1 \leq -0.2$.

For each vertex of the process parameter set $\mathcal{L}$, it should be checked whether the optimal controller χ_O positions the polynomial coefficients $\mathbf{p}$ within

the allowed pole region ρ_A . If vertex p_i would be outside ρ_A , the nearest vertex p_{Ai} of ρ_A should be determined. Then a new controller χ needed to be found such that for the related parameter polytope vertex θ_i , the new vertex p_i would equal p_{Ai} . As this may place another vertex outside the allowed range, the check must be repeated.

To simplify this procedure, the set ρ_A was transformed into an allowed controller set $\mathcal{I}$ for the process parameter set $\mathcal{L}$. Of this allowed controller set, the vertex was used which is nearest to the optimal controller χ_O outside this set. As this places all vertices in the allowed range, no repetition of this check is needed.

For the example with initial parameter uncertainty (5) and the desired polynomial $P_D(q^{-1})$, control design (14) after correction yields the controller $\tau_1 = 1$, $s_0 = 37.3$, $t_0 = 57.9$. Figure 2 shows the corresponding closed-loop poles for all vertices.

Fig. 2. Initial pole locations X with desired pole indicated by O

Obviously, all of these poles are within the allowed region. The initial controller was used to control all vertex processes (5) for a block wave reference signal. The responses for the vertex processes 1 and 4 yielding the slowest and fastest poles respectively are shown in Fig. 3.

Fig. 3. Response to block wave for initial robust pole-placer
a) Slowest poles; b) Fastest poles

The slowest and fastest processes correspond to vertex 1 and 4 respectively. The results show a large steady-state error and large difference in rise time.

3.4. Integrating Optimal Robust Pole-Placer

If at any time the steady-state error described above is not acceptable, it can be eliminated by introducing an integrator in the feedback by multiplying $R(q^{-1})$ by a factor $(1 - q^{-1})$, which instead of (9) yields the Diophantine equation for the integrating controller

$$P(q^{-1}) = (1 - q^{-1})R(q^{-1})A(q^{-1}) + S(q^{-1})B(q^{-1}). \quad (15)$$

Thus the order of the closed-loop polynomial $P(q^{-1})$ is increased by 1 relative to (9), and hence positioning of all poles requires the degree of $S(q^{-1})$ also to be increased by 1. By introducing the integrator part into the Sylvester matrix $\mathbf{M}(\theta)$, the same algorithm is applicable.

As in this case for the example the poles may be complex, the allowed region for the poles of $P(q^{-1})$ has to be specified in the complex plane. To assure reasonable rise and settling times, damping and control amplitude, the three vertices of an allowed polytope in the coefficients p_i were chosen which correspond to the allowed region for the poles of $P(q^{-1})$ shown in Fig. 4.

Fig. 4. Allowed region for complex closed-loop poles

To obtain minimal rise time, the desired poles were chosen at $z = 0$. Like the non-integrating controller, the resulting integrating controller was used to control all vertex processes (5) for a block wave reference signal. The slowest and fastest responses are shown in Fig. 5.

It can be seen from this figure, that the steady-state error is removed at the cost of a lower rise time.

Fig. 5. Response to block wave for initial integrating robust pole-placer: a) Slowest poles; b) Fastest poles

4. GUARANTEED ESTIMATION AND RESULTING FINAL CONTROL

4.1. Open-Loop Estimation

The parameter estimator used is similar to the one described by Veres and Norton (1991), which is used here in closed loop and for adaptive robust control.

This estimator for each sample k determines the parameter set L for which feasible parameter values a and b follows from the ARX model, true parameters a and b and disturbance bound δ as

$$\begin{aligned} |(b - l)u(k-1) - (a - \alpha)y(k)| &\leq \delta - |e(k)| \Rightarrow \\ |b - l|_{\alpha=a} &\leq \frac{\delta - |e(k)|}{|u(k-1)|}; \quad a - \alpha|_{l=b} \leq \frac{\delta - |e(k)|}{|y(k-1)|}, \end{aligned} \quad (16)$$

so that the convergence increases for larger amplitude of all signals. This instantaneous parameter set L intersects with the a-priori parameter set $\bar{L}$ to form an a-posteriori parameter set $\hat{L}$. To check the behavior of this estimator for the process used, it was applied in open loop for a uniformly distributed uncorrelated test sequence $u(k)$. For the nominal process, the resulting bounds on a and b are shown in Fig. 6.

It can be seen from Fig. 6, that the uncertainty in b decreases faster than that in a . This must be due to the large initial uncertainty in b relative to that in a .

4.2. Closed-Loop Estimation

If the process cannot be left uncontrolled, the robust controller designed from the initial process parameter range should be applied, and the process must be identified in closed loop with the feedback.

Fig. 6. Open loop identification of nominal process:
a) Test signal; b) Output signal;
c) Bounds on parameter b ; d) Bounds on parameter a ;
e) Final parameter region with x the nominal parameters

Van de Waal (1993) showed that for constant reference and controller, the uncertainty region does not decrease in a specific direction determined by the controller. Hence for fast convergence, the initial controller was applied to the process with reference signal u_C a white sequence uniformly distributed between ± 1 . The resulting final uncertainty region after 100 samples is shown in Fig. 7.

Fig. 7. Final closed-loop uncertainty region with nominal parameters indicated by X

4.3. Final Optimal Robust Pole-Placer Performance

When the final closed-loop estimated parameter uncertainty of Fig. 7 is used for the control design (14), the resulting pole locations are shown in Fig. 8.

Fig. 8. Final pole locations X with desired pole indicated by O

Evidently these final poles are much closer together than the initial ones. Like the initial non-integrating controller, the resulting final non-integrating controller was used to control all vertex processes (5) for a block wave reference signal. The responses for the vertex processes 3 and 6 yielding the slowest and fastest poles respectively are shown in Fig. 9.

Fig. 9. Response to block wave for final non-integrating robust pole-placer: a) Slowest poles; b) Fastest poles

Figure 9 shows that although the rise time variance and steady-state errors are lower than the initial ones, the steady-state error is still substantial. Therefore also here integral control was used. For large parameter uncertainty, this yielded reasonable average performance. For small parameter uncertainty, the poles tended to cluster in the extreme left corners of the allowed pole region of Fig. 4, causing high overshoot. A remedy for this problem is described in the next chapter.

If time-variance is suspected or to detect faults, it is worthwhile to check if the process is outside the a-priori uncertainty, that is if the process parameter set $\hat{Z}$ is empty. In that case appropriate measures should be taken.

5. SELF-TUNING ROBUST CONTROL

It is straightforward to combine the closed-loop guaranteed estimator and the design of a controller at each sample to form a self-tuning optimal robust pole-placement controller. For the non-integrating controller, Fig. 10 shows the output, input and controller parameters with the non-integrating controller for process parameters equal to vertex 4 of the initial uncertainty set. It can be seen that the controller speedily converges to good performance, but initially shows steady-state errors.

Fig. 10. Self-tuning non-integrating control of vertex 4 process. a) Reference (dotted) and output (drawn) signal. b) Control signal. c) Feedback gain s_0 with nominal value (dotted). d) Feedforward gain t_0 with nominal value

To prevent the high overshoot for the integrating controller described in the previous section, a new desired polynomial was chosen as $P_D(q^{-1}) = (1 + (0.5 + 0.3i)q^{-1})(1 + (0.5 - 0.3i)q^{-1})$. For the resulting self-tuning integrating pole-placement controller, Fig. 11 shows the same items as Fig. 10. It can be seen that the initial steady-state errors are removed at the cost of slower responses.

6. CONCLUSIONS

The results show that even for the initial parameter uncertainty, the optimal robust pole-placer with minimal average steady-state error over the vertices yields reasonable performance. Introduction of an integrator eliminates the steady-state error at the cost

of a lower bandwidth. Combination with the efficient polytope estimator in closed loop was shown to yield an easy to implement self-tuning robust controller for practical processes with good performance and fast convergence.

Fig. 11. Self-tuning integrating control of vertex 4 process a) - c) like Fig. 10; d) Feedback parameter s_1

ACKNOWLEDGMENT

The authors thank Dr. H.M. Falkus for the use of his MATLAB files for efficient polytope estimation.

REFERENCES

- Kuntzevich, V.M. and M. Lychak (1992). *Guaranteed estimates, adaptation and robustness in control systems*. Springer, Berlin.
- Lönnberg, P. and J. van Amerongen (1989). Input-constrained self-tuning control of a robot link. *Proc. IFAC symp. on Adaptive Systems in Control and Signal Processing*. Pergamon, Oxford, II, 329-334.
- Van de Waal, E.H. (1993). *Parameter-adaptive robust pole-placement control*. Report 93R080, Control Systems and Computer Engineering Group, Dept. of Electrical Engineering, University of Twente.
- Veres, S.M. and J.P. Norton (1991). Adaptive pole-placement control using parameter bounds. *Proceedings 30th Conference on Decision and Control*, Brighton, England. December, 2860-2865.