A Candidate to Replace PID Control: SISO Constrained LQ Control[1]

James B. Rawlings
Department of Chemical Engineering
University of Wisconsin–Madison

Austin, Texas
February 9, 2004

[1] This talk is based on [7, 8] by Pannocchia, Laachi and Rawlings

Outline

- Motivation. PID and Model-based control
- Proposal. CLQ control
- Examples and comparison
- Conclusions

Motivation: six myths of PID and LQ control

Myth 1: A PID controller is simpler to implement and tune than an LQ controller.

Myth 2: A PID controller with model-based tuning is as good as model-based control for simple processes such as SISO, 1st order plus time delay.

Myth 3: A well-tuned PID controller is more robust to plant/model mismatch than an LQ controller.

Myth 3 (alternate version): LQ controllers are not very robust to plant/model mismatch

Motivation: six myths of PID and LQ control

Myth 4: Integrating the tracking error as in PID control is necessary to remove steady-state offset. Applying some anti-windup strategy for this integrator is therefore necessary when an input saturates.

Myth 5: For simple processes (SISO, 1st order plus time delay) in the presence of input saturation, a PID controller with a simple anti-windup strategy is as good as model predictive control.

Myth 6: PID controllers are omnipresent because they work well on most processes.

Introduction

- PID control for single-input single-output (SISO) systems shows up everywhere in chemical process applications and process control education.

- Tuning rules are presented in numerous texts and, surprisingly, remain a topic of current control research [1, 10].

- *Question:* is PID's popularity due to any concrete technological advantage?

Technical advantages ascribed to PID control

- PID is simple, fast, and easy to implement in hardware and software
- PID is easy to tune
- PID provides good nominal control performance
- PID is robust to model errors

Model-based Control

- Model-based control methods include linear quadratic (LQ) control of unconstrained systems, and model predictive control (MPC) of constrained systems
- MPC is regarded by many in process control as complex to implement and tune
- The robustness of LQ control to model error has been a topic of debate [2]
- Some claim that PID controllers outperforms MPC controllers in the rejection of unmeasured load disturbances [9]

MPC's success in applications

- MPC has become the advanced controller of choice by industry mainly for the economically important, large-scale, multivariable processes in the plant

- The rationale for MPC in these applications is that the complexity of implementing MPC is justified only for the important loops with large payoffs

A modest proposal

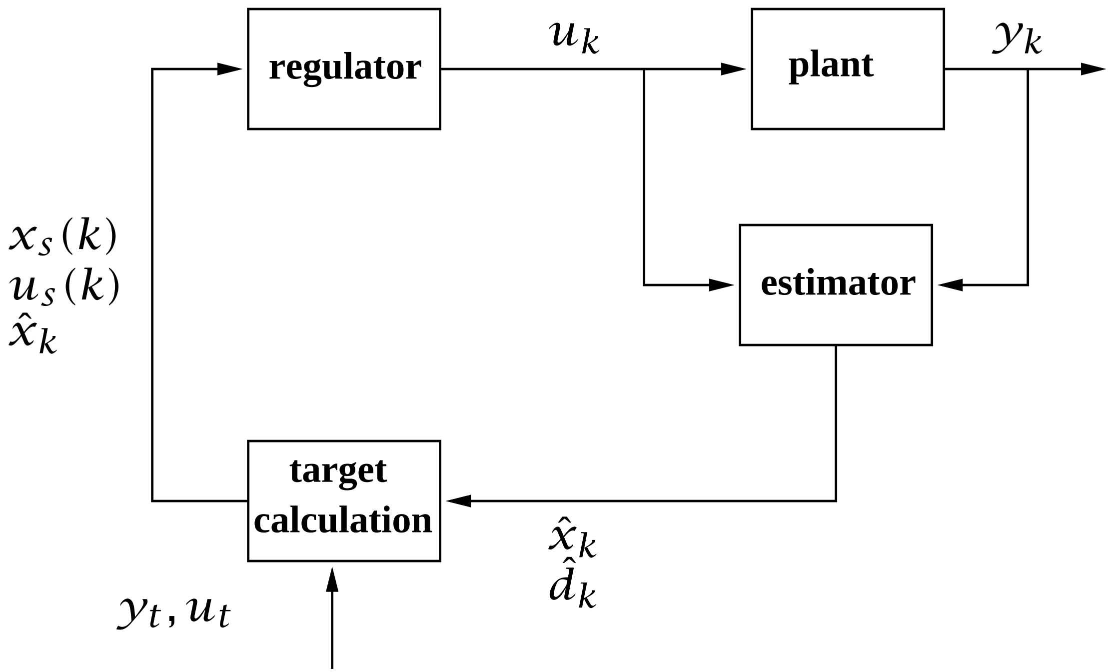
- To address this perception of complexity, we propose a constrained, SISO linear quadratic controller (CLQ) with the following features:

- CLQ is essentially as fast to execute as PID (within a factor of five regardless of system order)

- CLQ is easy to implement in software and hardware

- CLQ displays both higher performance and better robustness than PID controllers

CLQ: regulation, estimation and steady-state targets



Tuning parameters

- Regulator: output/input penalty, Q/S.
- Estimator: disturbance variance/measurement variance, Q_d/R_v.

Implementation of CLQ

- Model: System *plus* disturbance to remove offset [4, 6, 5]

$$\begin{bmatrix} x \\ d \end{bmatrix}_{k+1} = \begin{bmatrix} A & B \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix}_k + \begin{bmatrix} B \\ 0 \end{bmatrix} u_{k-m}$$

$$y_k = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix}_k$$

- Estimator: unconstrained Kalman filter (fast)

$$\hat{x}_{k+1} = A\hat{x}_k + L_x(y_k - C\hat{x}_k)$$

$$\hat{d}_{k+1} = \hat{d}_k + L_d(y_k - C\hat{x}_k)$$

- Target: analytical solution for SISO case (fast)

- Regulator: the expensive part

The regulator QP

Let $v = \{v_0, v_1, \ldots, v_{N-1}\}$ be the sequence of inputs. We can write the regulator as a strictly convex QP:

$$\min_v \frac{1}{2} v^T H v + v^T c \tag{1a}$$

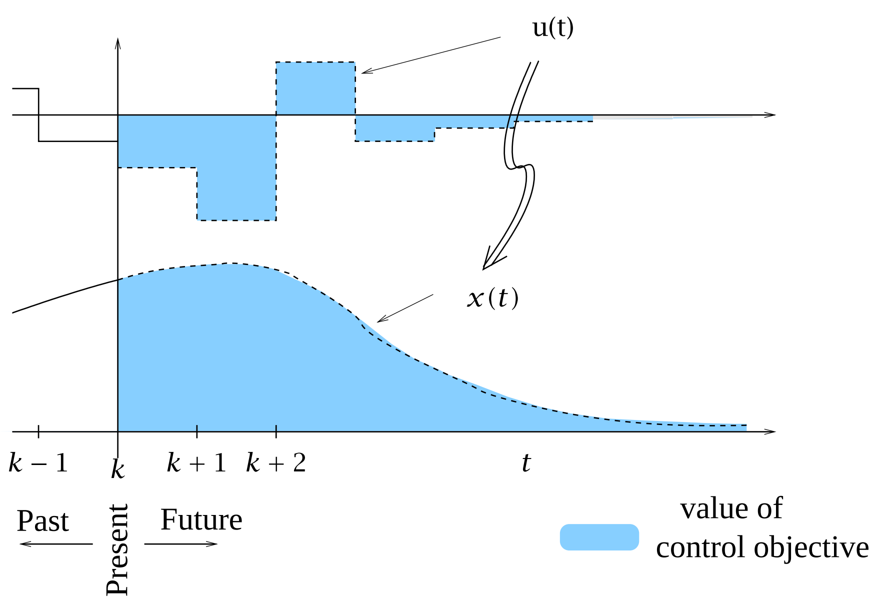
subject to:

$$\begin{bmatrix} \underline{u} \\ \underline{u} \\ \vdots \\ \underline{u} \end{bmatrix} \leq \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_{N-1} \end{bmatrix} \leq \begin{bmatrix} \overline{u} \\ \overline{u} \\ \vdots \\ \overline{u} \end{bmatrix} \tag{1b}$$

Let $v^* = (v_0^*, \ldots, v_{N-1}^*)$ denote the optimal solution to (1). The current control input is

$$u_k = u_s(k) + v_0^*$$

Storing the active sets of the regulator



- Each control move u_k can be at the upper bound, at the lower bound, or somewhere in between.

- We construct all combinations of constraints, 3^N.

- For $N = 4$, $3^4 = 81$ different active sets.

Regulator implementation — Two basic steps

1. The offline generation of a solution table. This step involves solving linear equations, multiplications and additions.

2. The online table scanning given the current value of x. This step involves only multiplications and additions and checking conditionals. These same operations are required in PID control.

The active set table

$$u_0 = K_i x + b_i$$

$N = 2$

i	constraint set	K_i	b_i
1	$\{\overline{u}, \overline{u}\}$	0	$\overline{u}$
2	$\{\overline{u}, -\}$	0	$\overline{u}$
3	$\{\overline{u}, \underline{u}\}$	0	$\overline{u}$
4	$\{-, \overline{u}\}$	K_4	b_4
5	$\{-, -\}$	K_5	b_5
6	$\{-, \underline{u}\}$	K_6	b_6
7	$\{\underline{u}, \overline{u}\}$	0	$\underline{u}$
8	$\{\underline{u}, -\}$	0	$\underline{u}$
9	$\{\underline{u}, \underline{u}\}$	0	$\underline{u}$

$N = 4$

i	constraint set	K_i	b_i
1	$\{\overline{u}, \overline{u}, \overline{u}, \overline{u}\}$	0	$\overline{u}$
2	$\{\overline{u}, \overline{u}, \overline{u}, -\}$	0	$\overline{u}$
3	$\{\overline{u}, \overline{u}, \overline{u}, \underline{u}\}$	0	$\overline{u}$
...	...	...	...
40	$\{-, -, -, \overline{u}\}$	K_{40}	b_{40}
41	$\{-, -, -, -\}$	K_{41}	b_{41}
42	$\{-, -, -, \underline{u}\}$	K_{42}	b_{42}
...	...	...	...
79	$\{\underline{u}, \underline{u}, \underline{u}, \overline{u}\}$	0	$\underline{u}$
80	$\{\underline{u}, \underline{u}, \underline{u}, -\}$	0	$\underline{u}$
81	$\{\underline{u}, \underline{u}, \underline{u}, \underline{u}\}$	0	$\underline{u}$

Example 1 — First order plus time delay

- The first example is a first order plus time delay (FOPTD) system:

$$G_1(s) = \frac{e^{-2s}}{10s + 1} \qquad \text{sampled with } T_s = 0.25$$

- The input is assumed to be constrained $|u| \leq 1.5$
- The control horizon is $N = 4$

Tuning

- The estimator is designed with $q_x = 0.05$ and $R_v = 0.01$ for both CLQ controllers

- The regulator input penalty is $s = 1$ for CLQ 1, and $s = 10$ for CLQ 2.

- The tuning parameters for PID 1 are chosen according to Luyben's rules [3, p. 97]: $K_c = 2.51$, $T_i = 17.3$, $T_d = 0$.

- The tuning parameters for PID 2 are chosen according to Skogestad's IMC rules [11]: $K_c = 2.35$, $T_i = 10$, $T_d = 0$.

Setpoint change and load disturbances

- In all simulations the setpoint is changed from 0 to 1 at time zero
- At time 25 a load disturbance passing through the same dynamics as the plant of magnitude -0.25 enters the system
- at time 50 the disturbance magnitude becomes -1 (which makes the setpoint 1 unreachable)
- finally at time 75 the disturbance magnitude becomes -0.25 again.

FOPTD system: nominal case.

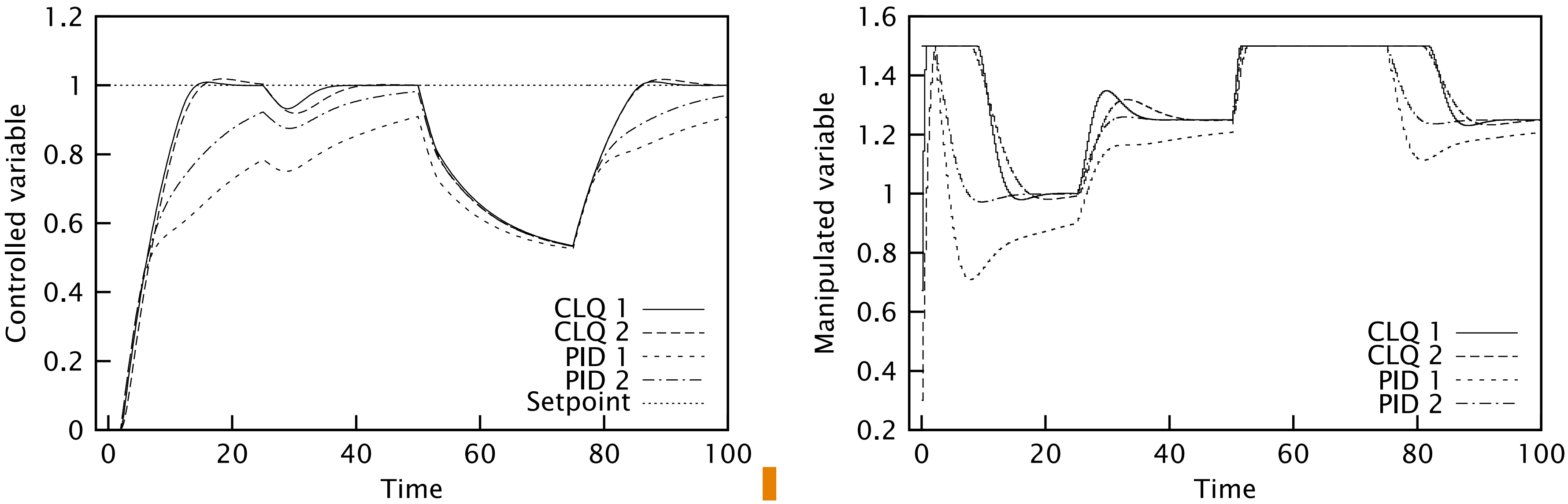


Figure 1: FOPTD system: nominal case.

FOPTD system: noisy case.

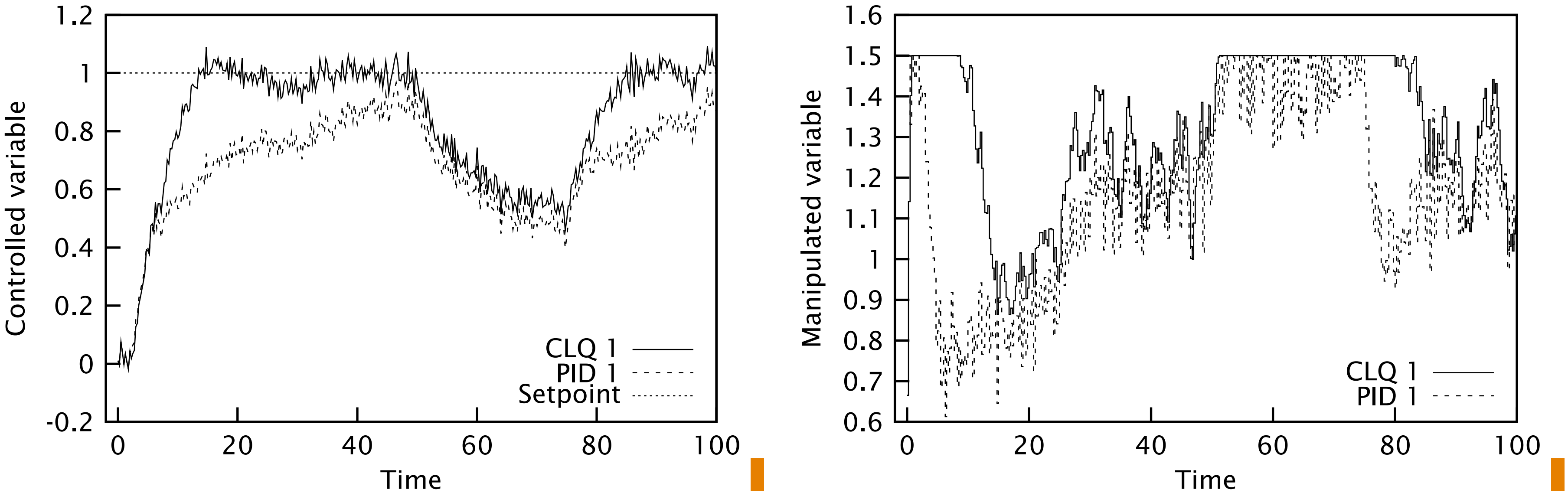


Figure 2: FOPTD system: noisy case.

FOPTD system: effect of plant/model mismatch.

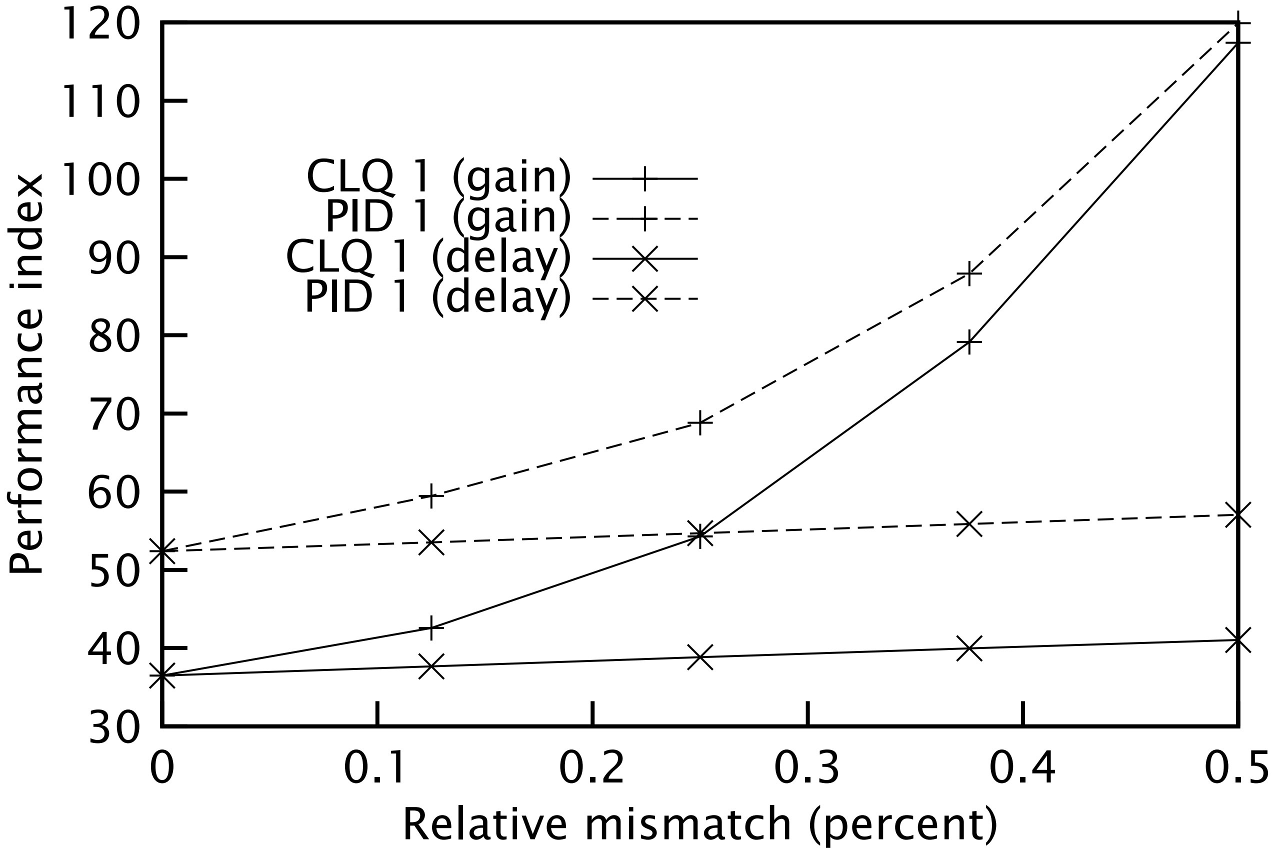


Figure 3: FOPTD system: effect of plant/model mismatch.

Example 2 — Integrating system

- The second example is an integrating system:

$$G_2(s) = \frac{e^{-2s}}{s} \qquad \text{sampled with } T_s = 0.25$$

- The same input constraints, horizon, setpoint change and disturbances, and estimator parameters as in the first example are considered.
- CLQ 1 uses a regulator input penalty of $s = 500$, while CLQ 2 uses $s = 5000$.
- The tuning parameters for PID 1 are chosen according to Luyben's rules [3, p. 97]: $K_c = 0.23$, $T_i = 18.7$, $T_d = 0$.
- The tuning parameters for PID 2 are chosen according to Skogestad's IMC rules [11]: $K_c = 0.23$, $T_i = 17$, $T_d = 0$.

Integrating system: nominal case.

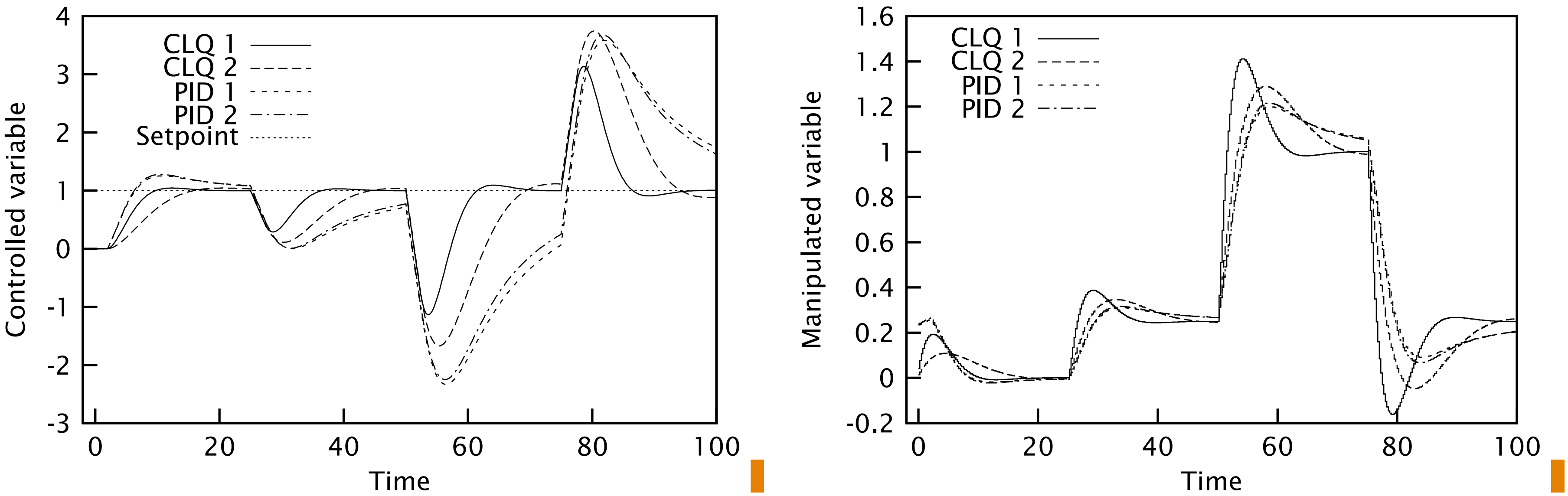


Figure 4: Integrating system: nominal case.

Integrating system: noisy case.

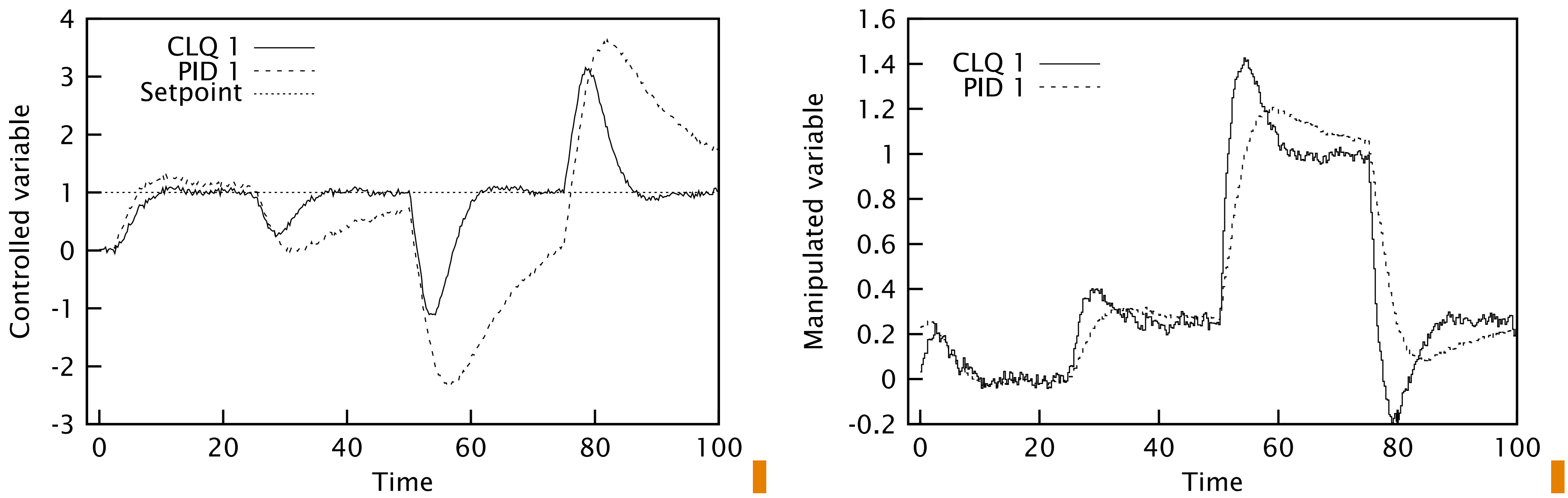


Figure 5: Integrating system: noisy case.

Integrating system: effect of plant/model mismatch.

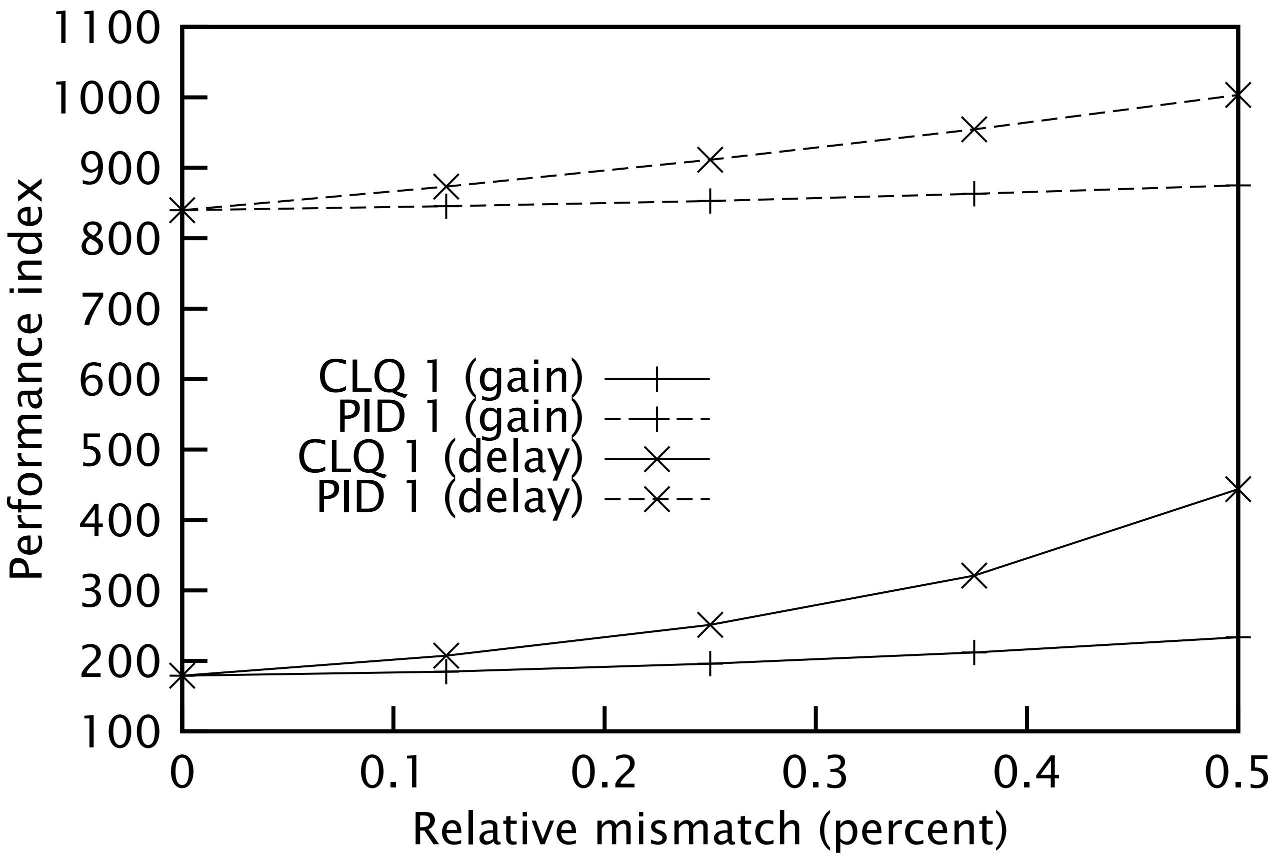


Figure 6: Integrating system: effect of plant/model mismatch.

Example 3 — Under-damped system

- The third example is a second-order, under-damped system:

$$G_3(s) = \frac{K}{\tau^2 s^2 + 2\tau\xi s + 1} \qquad T_s = 0.25, K = 1, \tau = 5, \xi = 0.2$$

- The same input constraints, horizon, setpoint change and disturbances, and estimator parameters as in the first example are assumed.
- CLQ 1 uses a regulator input penalty of $s = 1$, while CLQ 2 uses $s = 10$.
- The tuning parameters for PID 1 are chosen according to Luyben's rules [3, p. 97]: $K_c = 7.29$, $T_i = 16.8$, $T_d = 1.21$.
- The tuning parameters for PID 2 are chosen following the same IMC approach as in [11]: $K_c = 0.40$, $T_i = 2$, $T_d = 12.5$.

Under-damped system: nominal case.

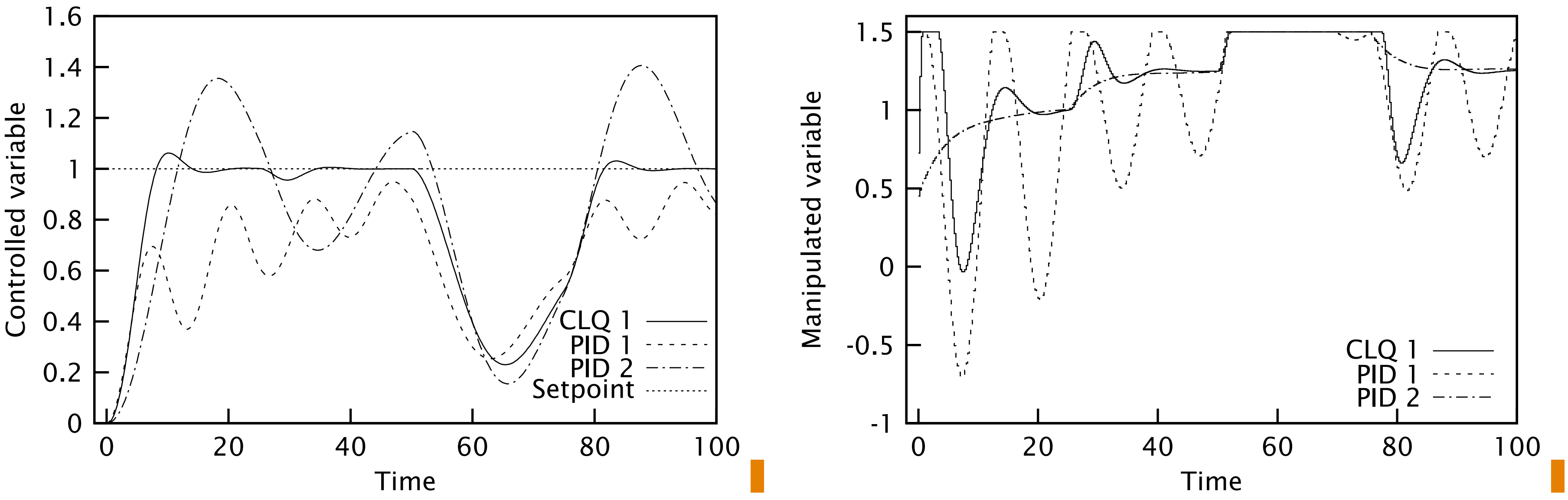


Figure 7: Under-damped system: nominal case.

Under-damped system: noisy case.

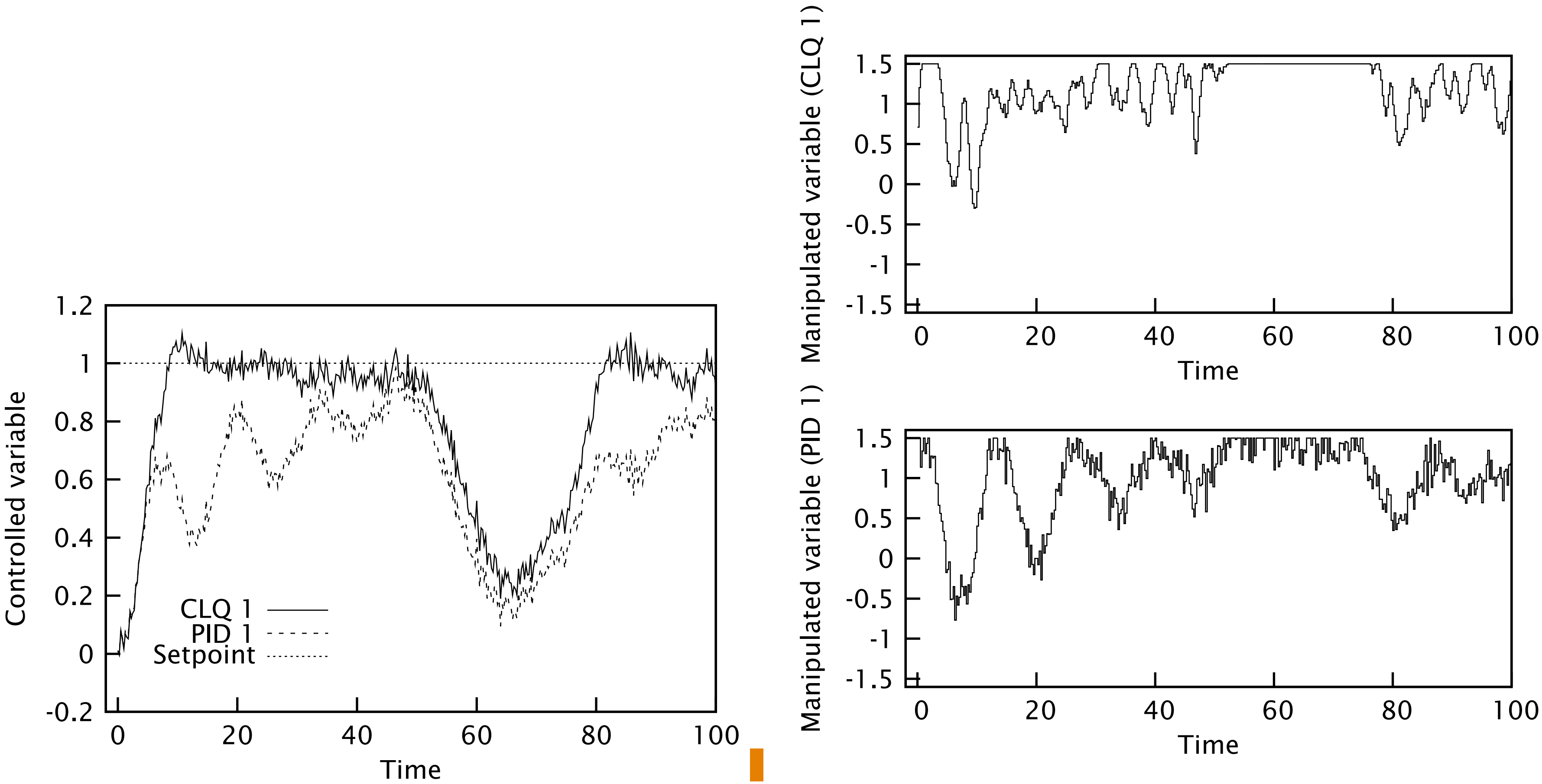


Figure 8: Under-damped system: noisy case.

Under-damped system: effect of plant/model mismatch.

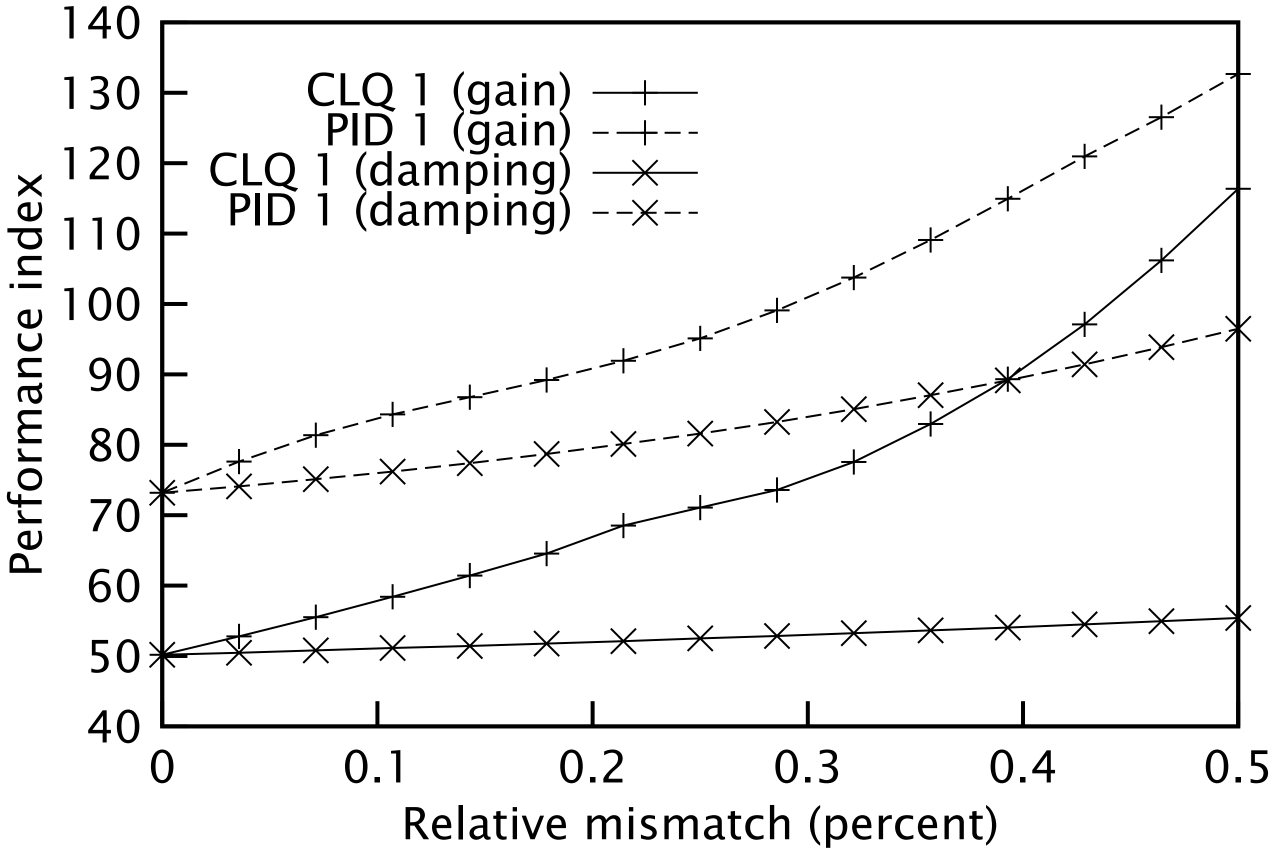


Figure 9: Under-damped system: effect of plant/model mismatch.

Computation time for CLQ

- The computational burden of CLQ is comparable to that of PID.

	average CPU time (ms)	maximum CPU time (ms)
PID	0.05	0.10
CLQ	0.22	0.55

- The CPU is a 1.7 GHz Athlon PC running Octave

Revisiting the six myths

Myth 1: A PID controller is simpler to implement and tune than an LQ controller.

- The validity of this myth rests largely with the hardware and control software vendors. Not difficult to implement CLQ if vendors offer on DCS.
- Regarding tuning, it is not difficult to look up tuning rules for a PID controller.
- It *is* difficult to find PID tuning parameters that give similar performance and robustness to an LQ controller.
- The LQ controller is not difficult to tune. The effects of its two tuning parameters are clear.

Myth 2: A PID controller with model-based tuning is as good as model-based control for simple processes such as SISO, 1st order plus time delay.

- No evidence to support this myth. Figure 1 shows the opposite is true.
- If we restrict "simple process" to "first-order process," this myth may remain in currency.

Myth 3: A well-tuned PID controller is more robust to plant/model mismatch than an LQ controller.

- No evidence to support this myth. Figures 3, 6, and 9 show the opposite is true.
- No superior robustness properties for PID control given any recommended tuning rules.

Myth 3 (alternate version): LQ controllers are not very robust to plant/model mismatch.

- One can construct processes for which the state feedback regulator has good margins but output feedback with the same regulator and a state estimator has poor margins [2].
- We have yet to see examples that indicate this issue has industrial significance.

Myth 4: Integrating the tracking error as in PID control is necessary to remove steady-state offset. Applying some anti-windup strategy for this integrator is therefore necessary when an input saturates.

- Integrating the tracking error is not required for offset free control as shown in all of the examples
- Integrating the model error is a sharper idea, and also removes the need for an anti-windup strategy when the input saturates.

Myth 5: For simple processes (SISO, 1st order plus time delay) in the presence of input saturation, a PID controller with a simple anti-windup strategy is as good as model predictive control.

- The constraint handling properties of PID are not competitive with MPC.
- Even for SISO, the difference can be noticeable. See Figures 1 and 7.

Myth 6: PID controllers are omnipresent because they work well on most processes.

- Seeing no evidence that PID controllers work particularly well, consider an explanation rooted more in human behavior.
- PID controllers are everywhere because vendors programmed them in the DCS when they replaced analog PID.

Disruptive Technology — Benefits of CLQ

- Provides a single, scalable control technology ranging from the fastest SISO loop to the slowest, largest, MIMO dynamic plant optimization
- Because of the model forecast, constraints and optimization features, we can network many SISO CLQs together to achieve full benefits of multivariable MPC control
- Take advantage of these “smart controllers” embedded at all plant levels
- Modify model used in forecast as conditions change
- Rewrite objective function to achieve changing plant objectives

Disruptive Technology — Costs of CLQ

- Vendor companies will have to implement on the DCS
- Modest modeling cost (SISO step test)
- Operators will need new training
- Textbook materials will need to be revised
- Inertia will have to be overcome

Acknowledgments

- Gabriele Pannocchia, University of Pisa (coauthor)
- Nabil Laachi, Ecole des Mines de Saint-Etienne (coauthor)
- Drs. Badgwell, Brambilla, Downs and Ogunnaike
- Drs. Edgar and Qin
- John Eaton for CLQ coding advice
- Financial support from TWMCC members and NSF grant #CTS-0105360

References

[1] D. Chen and D. E. Seborg. PID controller design based on direct synthesis and disturbance rejection. In *AIChE Annual Meeting*, page paper 276i, Reno (NV), USA, November 2001.

[2] J. C. Doyle. Guaranteed margins for LQG regulators. *IEEE Trans. Auto. Cont.*, 23:756–757, 1978.

[3] W. L. Luyben and M. L. Luyben. *Essentials of Process Control*. McGraw-Hill Int. Editions, 1997.

[4] K. R. Muske and T. A. Badgwell. Disturbance modeling for offset-free linear model predictive control. *J. Proc. Cont.*, 12:617–632, 2002.

[5] G. Pannocchia. Robust disturbance modeling for model predictive control with application to multivariable ill-conditioned processes. *J. Proc. Cont.*, 13:693–701, 2003.

[6] G. Pannocchia and E. C. Kerrigan. Offset-free control of constrained linear discrete-time systems subject to persistent unmeasured disturbances. In *42nd IEEE Conference on Decision and Control*, Hawaii, USA, December 2003.

[7] G. Pannocchia, N. Laachi, and J. B. Rawlings. A candidate to replace PID control: SISO constrained LQ control. Submitted for publication in *AIChE J*, 2004.

[8] G. Pannocchia, N. Laachi, and J. B. Rawlings. A fast, easily tuned, SISO, model predictive controller. In *DYCOPS*, Boston, MA, July 2004.

[9] F. G. Shinskey. *Feedback Controllers for the Process Industries*. McGraw-Hill, Inc, 1994.

[10] S. Skogestad. Probably the best simple PID tuning rules in the world. In *AIChE Annual Meeting*, page paper 276h, Reno (NV), USA, November 2001.

[11] S. Skogestad. Simple analytic rules for model reduction and PID controller tuning. *J. Proc. Cont.*, 13:291–309, 2003.