Predictive Control
Computer Exercise 1

Department of Automatic Control
Lund Institute of Technology

This is a simulation exercise in FRTN15 Predictive Control. It should give you an introduction to adaptive control, in particular Model Reference Adaptive Systems (MRAS). The systems are modeled and simulated in the Matlab/Simulink environment. For those not familiar with Matlab/Simulink, this exercise will also serve as an opportunity for you familiarise yourself with the software. The more you experiment and ask, the more you learn.

1. Feed-forward adaption with MIT-rule

Let $p = d/dt$ and consider a stable SISO process, $y(t) = G(p)u(t)$, with

$$G(p) = \frac{k}{p+1} \tag{1}$$

for some *unknown* $k > 0$, and a reference model, $y_m = G_m(p)u_c$, with

$$G_m(p) = \frac{k_0}{p+1}, \tag{2}$$

for some *known* parameter $k_0 > 0$. The problem is to construct a feedback $u(t)$ for the process, which makes its output $y(t)$ behave as the output $y_m(t)$ does when controlled by $u_c(t)$. If such a feedback can be established, then the unknown process may be controlled with an outer feedback law designed for the reference model. Clearly, if k is known, the problem can be solved by a simple proportional controller, letting

$$u(t) = \theta(t)u_c(t), \tag{3}$$

with $\theta(t) = k_0/k$ for all times, then

$$y(t) = G_m(p)u(t) = G_m(p)\theta(t)u_c(t) = G_m(p)u_c(t) = y_m(t) \tag{4}$$

However, if the gain of the process is unknown, we must find a way of adaptively choosing the parameter $\theta(t)$.

1.1 Adapting feed-forward gain with the MIT-rule

The most intuitive way of accomplishing the model matching, with $y(t) \to y_m(t)$ as $t \to \infty$, is to define an error between the process response and the reference model response and attempt to minimise it. Let

$$e(t) \triangleq y(t) - y_m(t) = G(p)\theta(t)u_c(t) - G_m(p)u_c(t). \tag{5}$$

In order to minimise this metric, we define a a positive definite function $\mathcal{E}(e) \triangleq \frac{1}{2}e^2$ of the error, whose time-derivative takes the form

$$\frac{d\mathcal{E}(e)}{dt} = \frac{1}{2}\frac{d\mathcal{E}(e)}{de}\frac{de}{d\theta}\frac{d\theta}{dt} = e\frac{de}{d\theta}\frac{d\theta}{dt}. \tag{6}$$

By defining a feedback law with some constant $\alpha > 0$ and

$$\frac{d\theta(t)}{dt} \triangleq -\alpha\Big(e\frac{de}{d\theta}\Big) \Rightarrow \frac{d\mathcal{E}(e)}{dt} = -2\alpha\Big(e\frac{de}{d\theta}\Big)^2 \leq 0, \tag{7}$$

the error will decrease with time, with $\mathcal{E}(e) \leq \mathcal{E}(e(t_0))$. Updating the parameter estimate in this way is known as the *MIT-rule*. Furthermore, the partial derivative of $e(t)$ with respect to θ becomes,

$$\frac{de}{d\theta} = \frac{d}{d\theta}(G(p)\theta(t)u_c(t) - G_m(p)u_c(t)) \tag{8}$$

$$= G(p)u_c(t) = \frac{k}{k_0}G_m(p)u_c = \frac{k}{k_0}y_m(t) \triangleq \beta y_m(t) \tag{9}$$

for some constant $\beta > 0$ since $k, k_0 > 0$. The constant β is not known, but positive and constant. Therefore, by defining $\gamma \triangleq \alpha\beta > 0$, we end up with

$$\frac{d\theta(t)}{dt} = -\alpha\Big(e\frac{de}{d\theta}\Big) = -\alpha\beta e(t)y_m(t) = -\gamma e(t)y_m(t) \tag{10}$$

As $y_m(t)$, $y(t)$ and $\theta(t)$ are known at all times, the feedback defined in (10) may be implemented as shown in the block diagram below (see Figure 1).

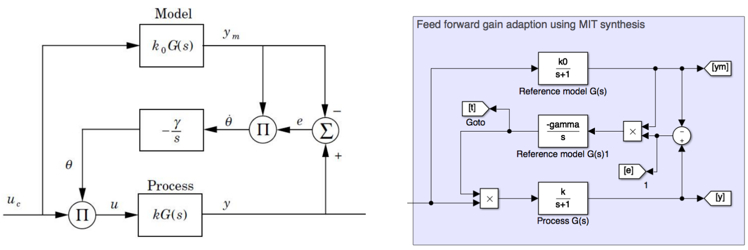

Figure 1 Block diagram (left) and functional Simulink implementation (right) of the MRAS feedforward gain adaption using the MIT-rule synthesis.

Exercise 1.1 In order to perform the simulations to verify and experiment with the theory, you will need to build a model of the system. Extract the .zip file and open the model `ex11.mdl`, which contains a reference generator, a set of blocks describing first order process and reference models, as well as functionality for plotting. It may be beneficial to connect the provided "goto"-ports to their corresponding signals - this uses the neat signal routing functionality Simulink for plotting.

1. *Implement the feedback control in Figure 1 in the* `ex11.mdl` *model.*
2. *Run the system with $\gamma = 1$, $k = 1$, $k_0 = 2$ to verify the implementation.*
3. *How does the rate of the adaption change with γ?*
4. *Does the parameter γ change the value to which θ converges?*

The method defining the parameter derivative is called the "MIT-rule", with

$$\frac{d\theta(t)}{dt} = -\gamma e(t)y_m(t), \tag{11}$$

but many other methods can be imagined. A common approach is the “Lyapunov-rule”, a method properly defined and presented in Section (2.2), whereby

$$\frac{d\theta(t)}{dt} = -\gamma e(t)u_c(t). \tag{12}$$

5. *Change the model so as to use the Lyapunov rule instead of the MIT-rule, do you see any difference?*

Further reading and useful notes The MIT-rule MRAS is derived in similar fashion here Åström and Wittenmark [2008] (see Example 5.1 Page 187). In addition, some nice and illustrative slides can be found here Freidovich [2010a], presenting a similar derivation to that of the book for the MIT-rule (available online for free). Note that the Lyapunov rule, which was only mentioned in brevity, may be done in many ways. However, special caution must be taken when using the state-space methods (see e.g. Page 212 in Åström and Wittenmark [2008]). Such methods typically invoke the KYP-lemma, only applying to strictly positive real (SPR) transfer functions (see Section 3), which is a very restrictive condition that only a handful of systems meet. A nice check for SPR is found in Lemma 6.1 on page 238 of Khalil [1996].

2. MRAS for a first order system

Again, let $p = d/dt$ and consider a SISO process, $y(t) = G(p)u(t)$, now with

$$G(p) = \frac{b}{p+a} \tag{13}$$

for some *unknown* $a, b > 0$, and a reference model, $y_m = G_m(p)u_c$, with

$$G_m(p) = \frac{b_m}{p+a_m}, \tag{14}$$

for some *known* parameter $a_m, b_m > 0$ and again consider the problem of model matching. Clearly, we now need to find two parameters, θ_1 and θ_2, to adapt both the gain and the pole location. For this purpose, consider a feedback law

$$u(t) = \theta_1(t)u_c(t) - \theta_2(t)y(t), \tag{15}$$

yielding a closed loop system

$$y(t) = \frac{b\theta_1}{p+a+b\theta_2}u_c(t) \tag{16}$$

Clearly, if we know the parameters $\{a, b\}$, then choosing

$$\theta_1(t) = \frac{b_m}{b} \triangleq \theta_1^0, \qquad \theta_2(t) = \frac{a_m - a}{b} \triangleq \theta_2^0, \tag{17}$$

yields $y = y_m(t)$, also known as perfect model following. However, in the case of unknown parameters $\{a, b\}$, the adaptive gains $\boldsymbol{\theta}(t) = [\theta_1(t), \theta_2(t)]^T$ need to be inferred just as in the feed-forward adaption in Section 1.1

2.1 Adaption by the MIT-rule

Just as in Section 1.1, consider an error $e(t) = y(t) - y_m(t)$ and define a positive definite error metic, in this case $\mathcal{E}(e) = \frac{1}{2}e^2$, which is to me minimised with respect to the adaptive gains. Again using the chain rule,

$$\frac{d\mathcal{E}(e)}{dt} = e\Big(\frac{\partial e}{\partial\theta_1}\frac{d\theta_1}{dt} + \frac{\partial e}{\partial\theta_2}\frac{d\theta_2}{dt}\Big). \tag{18}$$

The is very similar to the previous exercise, and choosing $\alpha_1, \alpha_2 > 0$,

$$\frac{d\theta_1}{dt} \triangleq -\alpha_1 e\frac{\partial e}{\partial\theta_1}, \quad \frac{d\theta_2}{dt} \triangleq -\alpha_2 e\frac{\partial e}{\partial\theta_2}, \tag{19}$$

yields a monotonically decreasing error $\mathcal{E}(e)$ with time. With (16), the partial derivatives needed to compute the parameter time-derivatives are

$$\frac{\partial e}{\partial\theta_1} = \frac{b}{p+a+b\theta_2}u_c(t), \tag{20}$$

$$\frac{\partial e}{\partial\theta_2} = -\frac{b^2\theta_1}{(p+a+b\theta_2)^2}u_c(t) = -\frac{b}{p+a+b\theta_2}y(t). \tag{21}$$

However, this presents a crucial problem, as the feedback requires knowledge of both a and b in order to be realised, which are *unknown* by the problem definition. One way around this issue is to assume that the adaptive gains are close to their optimal values in (17) at all times. If so,

$$\theta_2(t) \approx \frac{a_m - a}{b} \Leftrightarrow a + b\theta_2(t) \approx a_m, \tag{22}$$

implying that

$$\frac{\partial e}{\partial \theta_1} \approx \frac{b}{p + a_m} u_c(t), \qquad \frac{\partial e}{\partial \theta_2} \approx -\frac{b}{p + a_m} y(t). \tag{23}$$

As b is unknown but positive and constant, we simply define the adaptive gains as $\gamma_i \triangleq \alpha_i b / a_m$. The complete MRAS is then given by the feedback law in (15), where the parameters are updated by (19) using the approximation (23), written

$$\frac{d\theta_1}{dt} = -\gamma_1 \frac{a_m}{p + a_m} e(t) u_c(t), \qquad \frac{d\theta_2}{dt} = \gamma_2 \frac{a_m}{p + a_m} e(t) y(t). \tag{24}$$

Note especially the difference in sign, and that the approximation may be crude if starting far away from the true θ_i^0 parameters. In addition, it should be noted that the adaptive gains θ_i need not converge to their true values if the input signal to the system is not sufficiently exciting.

Exercise 1.2 Similar to the previous exercise, we need to build a model of the system in Simulink. Extract and the open the model `ex12.mdl`, containing a reference generator, a set of blocks describing first order (SISO) process and reference models, as well as functionality for plotting.

1. *Sketch the feedback on paper and then implement it in the* `ex12.mdl` *model.*
2. *Run with $\gamma = 1$, $a = 1$, $b = 0.5$, $a_m = b_m = 2$ to verify the implementation.*
3. *How does the rate of the adaption change with γ?*
4. *Do the parameters a and b affect the rate of adaption?*
5. *What happens when you alter the reference model?*
6. *Examine the parameter plane plot, with θ_2 as a function of θ_1. To which values do the parameters converge? Is this expected in theory?*

Further reading and notes The MIT-rule MRAS is derived in similar fashion in the book of Karl-Johan Åström Åström and Wittenmark [2008] (see Example 5.2 Page 190). In addition, some nice and illustrative slides can be found here Freidovich [2010b], which presents a similar derivation based on the same book (available online).

2.2 Adaption by the Lyapunov-rule

Another method which may be employed is to use conventional Lyapunov theory. To see this, start by writing the error dynamics using (14) and (16),

$$\frac{de(t)}{dt} = py(t) - py_m(t) \tag{25a}$$
$$= -(a + b\theta_2)y + b\theta_2 u_c - (-a_m y_m + b_m u_c) \tag{25b}$$
$$= -(a + b\theta_2)y + b\theta_2 u_c + a_m y_m - b_m u_c \tag{25c}$$
$$= -(a + b\theta_2)y + b\theta_2 u_c + a_m y - a_m y + a_m y_m - b_m u_c \tag{25d}$$
$$= -a_m e(t) - (b\theta_2(t) + a - a_m)y(t) + (b\theta_1(t) - b_m)u_c(t). \tag{25e}$$

Clearly, the objective is to drive the error to zero, but also to make the adaptive gain approach the values corresponding to perfect model following (17). As such, consider any Lyapunov function $\mathcal{V}(t) \geq 0$, where $\mathcal{V}(t) \to 0$ implies

- $e(t) \to 0$ (model following)
- $b\theta_1(t) - b_m \to 0$ (model matching numerator)
- $b\theta_2(t) + a - a_m \to 0$ (model matching denominator)

Choosing a Lyapunov function candidate is not always easy, but given the reasoning above, a suitable function could be chosen as

$$\mathcal{V}(t) = \frac{1}{2}\Big(e^2 + \frac{1}{b\gamma}(b\theta_2 + a - a_m)^2 + \frac{1}{b\gamma}(b\theta_1 - b_m)^2\Big). \tag{26}$$

with some gain $\gamma > 0$. Whereby application of the chain rule yields

$$\begin{aligned}\dot{\mathcal{V}}(t) =& e[-a_m e(t) - (b\theta_2(t) + a - a_m)y(t) + (b\theta_1(t) - b_m)u_c(t)]\\ &+ (b\theta_2 + a - a_m)^2 \frac{1}{b\gamma} b \frac{d\theta_1}{dt}\\ &+ (b\theta_1 - b_m)^2 \frac{1}{b\gamma} b \frac{d\theta_2}{dt}\\ &= -a_m e^2 + \frac{1}{\gamma}(b\theta_2 + a - a_m)\Big(\frac{d\theta_2}{dt} - \gamma y e\Big) + \frac{1}{\gamma}(b\theta_1 - b_m)\Big(\frac{d\theta_1}{dt} + \gamma u_c e\Big).\end{aligned}$$

Note that choosing

$$\frac{d\theta_1}{dt} \triangleq -\gamma u_c(t)e(t), \quad \frac{d\theta_2}{dt} \triangleq \gamma y(t)e(t), \tag{27}$$

yields a Lyapunov function time-derivative $\dot{\mathcal{V}}(t) = -a_m e(t)^2 \leq 0$. Indeed, one may show that $\ddot{\mathcal{V}}(t)$ is bounded implying that $\dot{\mathcal{V}}(t)$ is uniformly continuous, allowing a proof of global uniform asymptotic convergence of the error $e(t) \to 0$ as $t \to \infty$ by Theorem 4.8 in Khalil [1996], which is quite a bit stronger than the approximations done in the MIT-rule. However, it should be noted that much like the MIT-rule, the adaptive gains θ_i need not converge to their true values if the input signal to the system is not sufficiently exciting.

Exercise 1.3 Similarly to the previous exercise, we investigate the system in Simulink.

1. *Sketch the feedback on paper and then implement it in the* `ex13.mdl` *model.*
2. *Run with* $\gamma = 1$*,* $a = 1$*,* $b = 0.5$*,* $a_m = b_m = 2$ *to verify the implementation.*
3. *How does this feedback differ from that of the MIT-rule?*
4. *Which would you implement in practice? Which guarantees stability?*

Further reading and notes The Lyapunov-rule MRAS is derived in similar fashion but more extensively in the book of Karl-Johan Åström Åström and Wittenmark [2008] (see Example 5.7 Page 206). In addition, some nice and illustrative slides can be found here Freidovich [2010b] which, which presents a similar derivation (available online).

3. Notes on SPR transfer functions

The notion of a positive real (PR) transfer function, $G(s)$, is an intuitive but not too inclusive concept. By definition (see page 238 Khalil [1996]), a PR system satisfies

1. All poles of $G(s)$ are in $\Re\{s\} \leq 0$,
2. for all real $\omega \neq 0$ for which $i\omega$ is not a pole of $G(s)$, $G(i\omega)+G(-i\omega) > 0$,
3. any purely imaginary pole $i\omega$ of $G(i\omega)$ is simple (of multiplicity 1), and the residue $\lim_{s\to i\omega}(s - i\omega)G(s)$ is positive semidefinite hermitian.

This is clearly a very small subset of the set of stable transfer functions. Nonetheless, such systems exist, with many examples in circuit theory (see page 22. Brogliato et al. [2007]). An even smaller set of systems are strictly positive real (SPR), satisfying

4. $G(s + \epsilon)$ is positive real (PR) for some $\epsilon > 0$.

whereby the following relationship holds,

$$\text{SPR } G(s) \subset \text{ PR } G(s) \subset \text{Stable } G(s).$$

The conditions for SPR (along with conditions on observability and controllability) are necessary for the KYP-lemma to apply in the Lyapunov-rule synthesis by the state-space method used in (see e.g. Page 212 in Åström and Wittenmark [2008]). However, it may be possible to find suitable Lyapunov function candidates even for systems which are not SPR, but then there exist no standard method of synthesising the Lyapunov function.

References

Karl J Åström and Björn Wittenmark. *Adaptive control*. Dover, 2008.

Bernard Brogliato, Rogelio Lozano, Bernhard Maschke, and Olav Egeland. Dissipative systems analysis and control. *Theory and Applications*, 2, 2007.

Leonid Freidovich. Lecture 11: Model-Reference Adaptive Systems, 2010a. URL `http://www8.tfe.umu.se/forskning/Control_Systems/Courses/IterativeLearningAndAdaptiveControl/lectures/Lecture_11_EAILC.pdf`.

Leonid Freidovich. Lecture 12: Lyapunov-based design, 2010b. URL `http://www8.tfe.umu.se/forskning/Control_Systems/Courses/IterativeLearningAndAdaptiveControl/lectures/Lecture_12_EAILC.pdf`.

Mojtaba Hakimi-m and Hamid Khaloozadeh. Revision on the strict positive realness. In *IEEE, Int. Conf. ICTA'05, Greece, 15-16*, 2005.

Hassan K Khalil. Noninear systems. *Prentice-Hall, New Jersey*, 2(5):5–1, 1996.