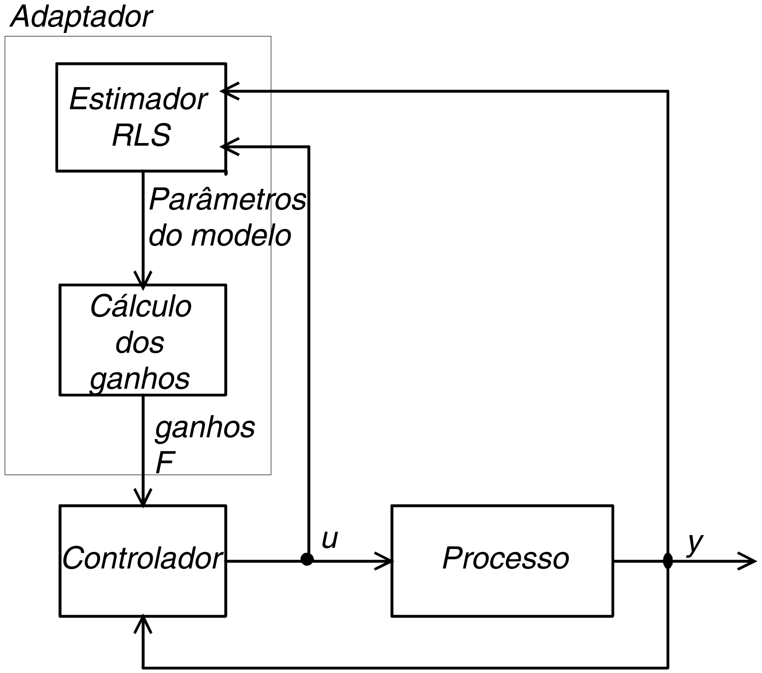
6. Controlo Preditivo e Controlo Adaptativo

Objectivo: *Mostrar como é possível integrar os blocos anteriormente estudados de identificação e controlo para obter controladores adaptativos.*

Referência: AW, Cap. 14

Controlador Adaptativo

Pode obter-se um controlador adaptativo acoplando um algoritmo de identificação recursivo a uma lei de controlo. São possíveis várias leis.



Controlo Adaptativo Autossintonizável

Processo real:

$$y(k+1) + ay(k) = u(k) + e(k+1) + ce(k)$$

Modelo:

$$y(k+1) + \theta y(k) = u(k) + v(k+1)$$

Lei de controlo de variância mínima em que o valor verdadeiro do parâmetro é substituído pela sua estimativa e se admite que o ruído é branco:

$$u(k) = \hat{\theta} y(k)$$

Repare-se que como o ruído **não** é branco, a estimativa de θ **não** converge para o valor "verdadeiro", que é igual a a .

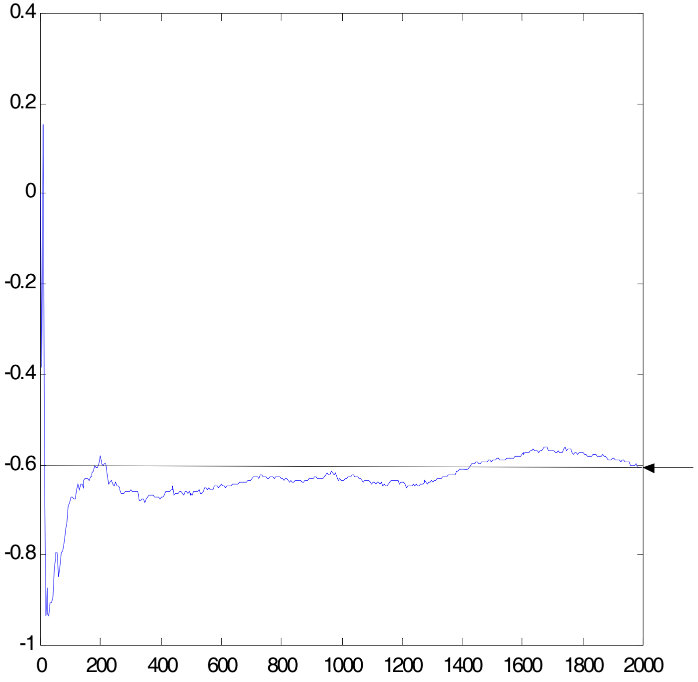
No entanto, o ganho do controlador (igual à estimativa de θ) aproxima-se do ganho óptimo dado por

$$F_{VM} = a - c$$

Por exemplo, se $a = -0.9$ e $c = -0.3$, o ganho "aproxima-se" de $F = -0.6$.

Quer dizer, embora as estimativas de mínimos quadrados dos parâmetros do modelo estejam polarizadas (devido ao ruído não ser branco), o ganho do controlador é ainda o ganho de variância mínima. É a esta propriedade que se dá o nome de **autossintonização** (*self-tuning*).

Convergência do ganho em presença de ruído colorido. A seta indica o valor do ganho de variância mínima.



Controlo adaptativo de um motor CC

Estrutura da função de transferência do sistema amostrado:

$$H(z) = \frac{K(z-b)}{(z-1)(z-a)}$$

Equação de diferenças correspondente:

$$y(t) = (1-q^{-1})(1-aq^{-1}) = K(1-bq^{-1})u(t-1)$$

Podemos tirar partido de sabermos que o sistema tem uma integração:

Definindo: $\Delta y(t) := (1-q^{-1})y(t)$ tem-se o modelo mais simples:

$$\Delta y(t) = a\Delta y(t-1) + Ku(t-1) - Kbu(t-2)$$

$$\Delta y(t) = a\Delta y(t-1) + Ku(t-1) - Kbu(t-2)$$

Definindo:

$$\varphi(t-1) := \left[\Delta y(t-1) \quad u(t-1) \quad u(t-2)\right]^T$$

$$\theta_o := \left[\theta_1 = a \quad \theta_2 = K \quad \theta_3 = -Kb\right]^T$$

O modelo escreve-se na forma de regressão linear:

$$\Delta y(t) = \varphi^T(t-1)\theta_o$$

Os parâmetros deste modelo podem assim ser facilmente estimados usando o algoritmo de estimação de mínimos quadrados recursivo (RLS).

A opartir das estimativas são calculados os ganhos do controlador de colocação de pólos.

Algoritmo Adaptativo Explícito de Posicionamento de Pólos

Em cada intervalo de amostragem, executar recursivamente os seguintes passos:

1. Calcular $\Delta y(t) = y(t) - y(t-1)$
2. Estimar por RLS o vector de parâmetros θ_o no modelo de regressão linear $\Delta y(t) = \varphi^T(t-1)\theta_o$
   a. Ganho de Kalman: $$\gamma(t) = \frac{P(t)\varphi(t-1)}{1+\varphi^T(t-1)P(t)\varphi(t-1)}$$
   b. $\hat{\theta}(t) = \hat{\theta}(t-1) + \gamma(t)\left[\Delta y(t) - \varphi^T(t-1)\hat{\theta}(t-1)\right]$

c. $P(t-1)=\left[I-\gamma(t)\varphi^T(t-1)\right]P(t)$

3. $\hat{a}=\hat{\theta}_1 \qquad \hat{K}=\hat{\theta}_2 \qquad \hat{b}=-\hat{\theta}_3/\hat{\theta}_2$

4. Ganhos do controlador (p_1 e p_2 são os coeficientes do polinómio característico desejado, $z^2+p_1z+p_2$):

a. $t_0=\dfrac{1+p_1+p_2}{\hat{K}} \qquad s_0=\dfrac{1+\hat{a}+p_1}{\hat{K}} \qquad s_1=\dfrac{p_2-\hat{a}}{\hat{K}}$

b. $r_1=-\hat{b}+\dfrac{\hat{b}(\hat{b}+p_1\hat{b}+p_2)}{(\hat{b}-1)(\hat{b}-a)}$ (sem cancelamento do zero)

5. Controlo a aplicar

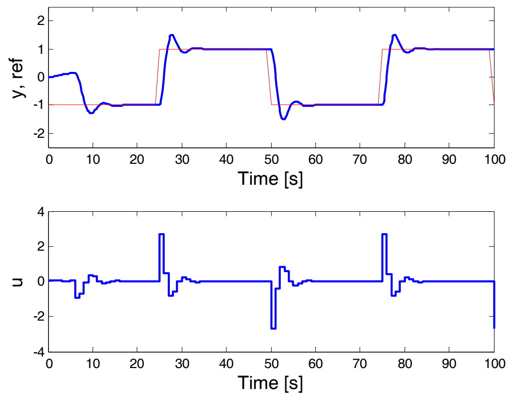
$$u(t) = t_0 u_c(t) - s_0 y(t) - s_1 y(t-1) - r_1 u(t-1)$$

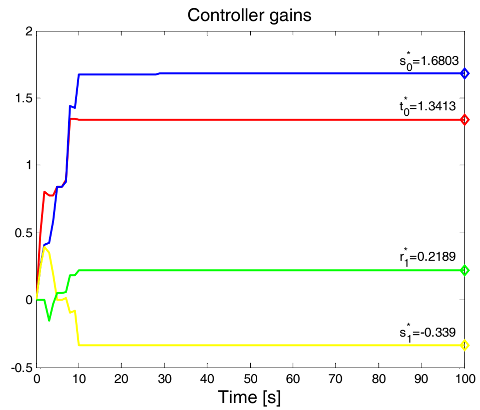
O estimador RLS é inicializado como

$$\hat{\theta}(0) = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T \qquad P(0) = 1000I$$

O valor inicial de $\hat{\theta}_2$ é escolhido diferenrte de zero (podia ser outro valor) para evitar a divisão por zero quando se projecta o controlo.

Results with no dither noise added to the manipulated variable





During the first 5 seconds the motor is in open loop.

During the first 10 seconds a low power dither noise is added to the manipulated variable, in order to improve the convergence of RLS.