

Traitement Statistique et Approche Système

...

TD4 d'Approche Système

...

Identification et Contrôle (par f^o de sensibilité)

Benoît Marx

Centre de Recherche en Automatique de Nancy (CRAN)
Ecole Nationale Supérieure de Géologie (ENSG)

1 Objectifs

- Données : entrée $x(t)$ et sortie $y(t)$ mesurées d'un système
fichier `dataSS4.mat` sur Arche

1 Objectifs

- Données : entrée $x(t)$ et sortie $y(t)$ mesurées d'un système
fichier `dataSS4.mat` sur Arche



Identification

minimisation de $J_i(\theta) = \int_0^{\infty} \frac{1}{2} (y_m(t, \theta) - y(t))^2 dt$



1 Objectifs

- Données : entrée $x(t)$ et sortie $y(t)$ mesurées d'un système
fichier `dataSS4.mat` sur Arche



Identification

minimisation de $J_i(\theta) = \int_0^{\infty} \frac{1}{2} (y_m(t, \theta) - y(t))^2 dt$



- Résultat : modèle du système $G_m(p)$ t.q. : $y_m(t) \approx y(t)$

1 Objectifs

- Données : entrée $x(t)$ et sortie $y(t)$ mesurées d'un système
fichier `dataSS4.mat` sur Arche



Identification

minimisation de $J_i(\theta) = \int_0^{\infty} \frac{1}{2} (y_m(t, \theta) - y(t))^2 dt$



- Résultat : modèle du système $G_m(p)$ t.q. : $y_m(t) \approx y(t)$



Contrôle

minimisation de $J_c(\theta_c) = \int_0^{\infty} \frac{1}{2} (y_{BF}(t, \theta_c) - y_s(t))^2 dt$



1 Objectifs

- Données : entrée $x(t)$ et sortie $y(t)$ mesurées d'un système fichier `dataSS4.mat` sur Arche



Identification

minimisation de $J_i(\theta) = \int_0^{\infty} \frac{1}{2} (y_m(t, \theta) - y(t))^2 dt$



- Résultat : modèle du système $G_m(p)$ t.q. : $y_m(t) \approx y(t)$



Contrôle

minimisation de $J_c(\theta_c) = \int_0^{\infty} \frac{1}{2} (y_{BF}(t, \theta_c) - y_s(t))^2 dt$

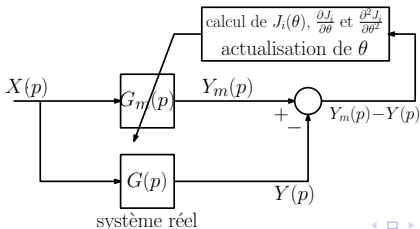


- Résultat : correcteur $C(p)$ t.q. : $y_{BF}(t) \approx y_s(t)$

2.a Identification

- On cherche un modèle du système de la forme : $G_m(p) = \frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1}$
- Trouver $\theta = [a_1 \ a_2 \ b_0 \ b_1]^T$ qui minimise le critère :

$$J_i(\theta) = \frac{1}{2} \int_0^{\infty} (y_m(t, \theta) - y(t))^2 dt$$



2.a Identification

- On cherche un modèle du système de la forme : $G_m(p) = \frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1}$
- Trouver $\theta = [a_1 \ a_2 \ b_0 \ b_1]^T$ qui minimise le critère :

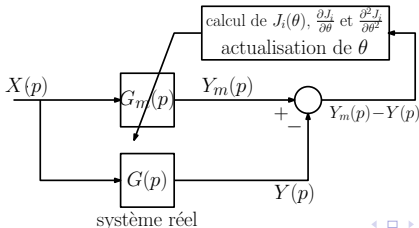
$$J_i(\theta) = \frac{1}{2} \int_0^{\infty} (y_m(t, \theta) - y(t))^2 dt$$

- Minimisation itérative par l'algorithme de **Newton-Raphson** :

$$\theta_{k+1} = \theta_k - \left(\frac{d^2 J_i(\theta_k)}{d\theta^2} \right)^{-1} \left(\frac{dJ_i(\theta_k)}{d\theta} \right)$$

- A chaque itération il faut calculer le **gradient** et le **hessien** du critère

$$G(\theta_k) = \frac{dJ_i(\theta_k)}{d\theta} \quad H(\theta_k) = \frac{d^2 J_i(\theta_k)}{d\theta^2}$$



2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^{\infty} (y_m(t, \theta) - y(t))^2 dt$
- Gradient (dérivée première) :

2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^{\infty} (y_m(t, \theta) - y(t))^2 dt$
- Gradient (dérivée première) :

$$\frac{dJ_i(\theta)}{d\theta} = \int_0^{\infty} \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) (y_m(t, \theta) - y(t)) dt$$

2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^\infty (y_m(t, \theta) - y(t))^2 dt$
- Gradient (dérivée première) :

$$\frac{dJ_i(\theta)}{d\theta} = \int_0^\infty \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) (y_m(t, \theta) - y(t)) dt$$

Approché par une somme de Riemann :

$$\frac{dJ_i(\theta)}{d\theta} \approx \sum_{n=1}^N \left(\frac{\partial y_m(t_n, \theta)}{\partial \theta} \right) (y_m(t_n, \theta) - y(t_n)) \delta t$$

2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^\infty (y_m(t, \theta) - y(t))^2 dt$
- Gradient (dérivée première) :

$$\frac{dJ_i(\theta)}{d\theta} = \int_0^\infty \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) (y_m(t, \theta) - y(t)) dt$$

Approché par une somme de Riemann :

$$\frac{dJ_i(\theta)}{d\theta} \approx \sum_{n=1}^N \left(\frac{\partial y_m(t_n, \theta)}{\partial \theta} \right) (y_m(t_n, \theta) - y(t_n)) \delta t$$

Sous forme matricielle :

$$\frac{dJ_i(\theta)}{d\theta} \approx \begin{bmatrix} \sigma_{a_1}(t_1) & \sigma_{a_1}(t_2) & \dots & \sigma_{a_1}(t_N) \\ \sigma_{a_2}(t_1) & \sigma_{a_2}(t_2) & \dots & \sigma_{a_2}(t_N) \\ \sigma_{b_0}(t_1) & \sigma_{b_0}(t_2) & \dots & \sigma_{b_0}(t_N) \\ \sigma_{b_1}(t_1) & \sigma_{b_1}(t_2) & \dots & \sigma_{b_1}(t_N) \end{bmatrix} \begin{bmatrix} y_m(t_1, \theta) - y(t_1) \\ y_m(t_2, \theta) - y(t_2) \\ \vdots \\ y_m(t_N, \theta) - y(t_N) \end{bmatrix} \delta t$$

2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^{\infty} (y_m(t, \theta) - y(t))^2 dt$
- Hessien (dérivée seconde) :

2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^\infty (y_m(t, \theta) - y(t))^2 dt$
- Hessien (dérivée seconde) :

$$\frac{d^2 J_i(\theta)}{d\theta^2} = \int_0^\infty \left(\frac{\partial^2 y_m(t, \theta)}{\partial \theta^2} \right) (y_m(t, \theta) - y(t)) + \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right)^T dt$$

2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^\infty (y_m(t, \theta) - y(t))^2 dt$
- Hessien (dérivée seconde) :

$$\frac{d^2 J_i(\theta)}{d\theta^2} = \int_0^\infty \left(\frac{\partial^2 y_m(t, \theta)}{\partial \theta^2} \right) (y_m(t, \theta) - y(t)) + \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right)^T dt$$

$$\frac{d^2 J_i(\theta)}{d\theta^2} \approx \int_0^\infty \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right)^T dt$$

2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^\infty (y_m(t, \theta) - y(t))^2 dt$
- Hessien (dérivée seconde) :

$$\frac{d^2 J_i(\theta)}{d\theta^2} = \int_0^\infty \left(\frac{\partial^2 y_m(t, \theta)}{\partial \theta^2} \right) (y_m(t, \theta) - y(t)) + \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right)^T dt$$

$$\frac{d^2 J_i(\theta)}{d\theta^2} \approx \int_0^\infty \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right)^T dt$$

Approché par une somme de Riemann :

$$\frac{d^2 J_i(\theta)}{d\theta^2} \approx \sum_{n=1}^N \left(\frac{\partial y_m(t_n, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t_n, \theta)}{\partial \theta} \right)^T \delta t$$

2.a et 2.b Calculs des gradient et hessien

- Critère : $J_i(\theta) = \frac{1}{2} \int_0^\infty (y_m(t, \theta) - y(t))^2 dt$
- Hessien (dérivée seconde) :

$$\frac{d^2 J_i(\theta)}{d\theta^2} = \int_0^\infty \left(\frac{\partial^2 y_m(t, \theta)}{\partial \theta^2} \right) (y_m(t, \theta) - y(t)) + \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right)^T dt$$

$$\frac{d^2 J_i(\theta)}{d\theta^2} \approx \int_0^\infty \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t, \theta)}{\partial \theta} \right)^T dt$$

Approché par une somme de Riemann :

$$\frac{d^2 J_i(\theta)}{d\theta^2} \approx \sum_{n=1}^N \left(\frac{\partial y_m(t_n, \theta)}{\partial \theta} \right) \left(\frac{\partial y_m(t_n, \theta)}{\partial \theta} \right)^T \delta t$$

Sous forme matricielle :

$$\frac{d^2 J_i(\theta)}{d\theta^2} \approx \begin{bmatrix} \sigma_{a_1}(t_1) & \dots & \sigma_{a_1}(t_N) \\ \sigma_{a_2}(t_1) & \dots & \sigma_{a_2}(t_N) \\ \sigma_{b_0}(t_1) & \dots & \sigma_{b_0}(t_N) \\ \sigma_{b_1}(t_1) & \dots & \sigma_{b_1}(t_N) \end{bmatrix} \begin{bmatrix} \sigma_{a_1}(t_1) & \sigma_{a_2}(t_1) & \sigma_{b_0}(t_1) & \sigma_{b_1}(t_1) \\ \sigma_{a_1}(t_2) & \sigma_{a_2}(t_2) & \sigma_{b_0}(t_2) & \sigma_{b_1}(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{a_1}(t_N) & \sigma_{a_2}(t_N) & \sigma_{b_0}(t_N) & \sigma_{b_1}(t_N) \end{bmatrix} \delta t$$

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

$$Y_m(p, \theta) = \left(\frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

$$Y_m(p, \theta) = \left(\frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

- Equation différentielle équivalente :

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

$$Y_m(p, \theta) = \left(\frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

- Equation différentielle équivalente :

$$a_2 \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_1 \frac{\partial y_m(t, \theta)}{\partial t} + y_m(t, \theta) = b_1 \frac{dx(t)}{dt} + b_0 x(t)$$

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

$$Y_m(p, \theta) = \left(\frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

- Equation différentielle équivalente :

$$a_2 \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_1 \frac{\partial y_m(t, \theta)}{\partial t} + y_m(t, \theta) = b_1 \frac{dx(t)}{dt} + b_0 x(t)$$

- Dérivées par rapport à a_1 , a_2 , b_0 et b_1 :

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

$$Y_m(p, \theta) = \left(\frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

- Equation différentielle équivalente :

$$a_2 \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_1 \frac{\partial y_m(t, \theta)}{\partial t} + y_m(t, \theta) = b_1 \frac{dx(t)}{dt} + b_0 x(t)$$

- Dérivées par rapport à a_1 , a_2 , b_0 et b_1 :

$$a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0$$

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

$$Y_m(p, \theta) = \left(\frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

- Equation différentielle équivalente :

$$a_2 \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_1 \frac{\partial y_m(t, \theta)}{\partial t} + y_m(t, \theta) = b_1 \frac{dx(t)}{dt} + b_0 x(t)$$

- Dérivées par rapport à a_1 , a_2 , b_0 et b_1 :

$$a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0$$

$$\frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0$$

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

$$Y_m(p, \theta) = \left(\frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

- Equation différentielle équivalente :

$$a_2 \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_1 \frac{\partial y_m(t, \theta)}{\partial t} + y_m(t, \theta) = b_1 \frac{dx(t)}{dt} + b_0 x(t)$$

- Dérivées par rapport à a_1 , a_2 , b_0 et b_1 :

$$a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0$$

$$\frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0$$

$$a_2 \frac{\partial^2 \sigma_{b_0}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_0}}{\partial t} + \sigma_{b_0} = x(t)$$

2.c Fonctions de sensibilité

- Relation entrée/sortie du modèle :

$$Y_m(p, \theta) = \left(\frac{b_1 p + b_0}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

- Equation différentielle équivalente :

$$a_2 \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_1 \frac{\partial y_m(t, \theta)}{\partial t} + y_m(t, \theta) = b_1 \frac{dx(t)}{dt} + b_0 x(t)$$

- Dérivées par rapport à a_1 , a_2 , b_0 et b_1 :

$$a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0$$

$$\frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0$$

$$a_2 \frac{\partial^2 \sigma_{b_0}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_0}}{\partial t} + \sigma_{b_0} = x(t)$$

$$a_2 \frac{\partial^2 \sigma_{b_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_1}}{\partial t} + \sigma_{b_1} = \frac{dx(t)}{dt}$$

2.d Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\left\{ \begin{array}{l} a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0 \\ \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0 \\ a_2 \frac{\partial^2 \sigma_{b_0}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_0}}{\partial t} + \sigma_{b_0} = x(t) \\ a_2 \frac{\partial^2 \sigma_{b_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_1}}{\partial t} + \sigma_{b_1} = \frac{dx(t)}{dt} \end{array} \right.$$

2.d Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\left\{ \begin{array}{l} a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0 \\ \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0 \\ a_2 \frac{\partial^2 \sigma_{b_0}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_0}}{\partial t} + \sigma_{b_0} = x(t) \\ a_2 \frac{\partial^2 \sigma_{b_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_1}}{\partial t} + \sigma_{b_1} = \frac{dx(t)}{dt} \end{array} \right.$$

- Fonctions de transfert des fonctions de sensibilités :

2.d Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\left\{ \begin{array}{l} a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0 \\ \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0 \\ a_2 \frac{\partial^2 \sigma_{b_0}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_0}}{\partial t} + \sigma_{b_0} = x(t) \\ a_2 \frac{\partial^2 \sigma_{b_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_1}}{\partial t} + \sigma_{b_1} = \frac{dx(t)}{dt} \end{array} \right.$$

- Fonctions de transfert des fonctions de sensibilités :

$$\mathcal{L}(\sigma_{a_1}(t, \theta)) = \Sigma_{a_1}(p, \theta) = \left(\frac{-p}{a_2 p^2 + a_1 p + 1} \right) Y_m(p, \theta)$$

2.d Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\left\{ \begin{array}{l} a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0 \\ \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0 \\ a_2 \frac{\partial^2 \sigma_{b_0}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_0}}{\partial t} + \sigma_{b_0} = x(t) \\ a_2 \frac{\partial^2 \sigma_{b_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_1}}{\partial t} + \sigma_{b_1} = \frac{dx(t)}{dt} \end{array} \right.$$

- Fonctions de transfert des fonctions de sensibilités :

$$\mathcal{L}(\sigma_{a_1}(t, \theta)) = \Sigma_{a_1}(p, \theta) = \left(\frac{-p}{a_2 p^2 + a_1 p + 1} \right) Y_m(p, \theta)$$

$$\mathcal{L}(\sigma_{a_2}(t, \theta)) = \Sigma_{a_2}(p, \theta) = \left(\frac{-p^2}{a_2 p^2 + a_1 p + 1} \right) Y_m(p, \theta)$$

2.d Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\left\{ \begin{array}{l} a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0 \\ \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0 \\ a_2 \frac{\partial^2 \sigma_{b_0}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_0}}{\partial t} + \sigma_{b_0} = x(t) \\ a_2 \frac{\partial^2 \sigma_{b_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_1}}{\partial t} + \sigma_{b_1} = \frac{dx(t)}{dt} \end{array} \right.$$

- Fonctions de transfert des fonctions de sensibilités :

$$\mathcal{L}(\sigma_{a_1}(t, \theta)) = \Sigma_{a_1}(p, \theta) = \left(\frac{-p}{a_2 p^2 + a_1 p + 1} \right) Y_m(p, \theta)$$

$$\mathcal{L}(\sigma_{a_2}(t, \theta)) = \Sigma_{a_2}(p, \theta) = \left(\frac{-p^2}{a_2 p^2 + a_1 p + 1} \right) Y_m(p, \theta)$$

$$\mathcal{L}(\sigma_{b_0}(t, \theta)) = \Sigma_{b_0}(p, \theta) = \left(\frac{1}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

2.d Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\left\{ \begin{array}{l} a_2 \frac{\partial^2 \sigma_{a_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_1}}{\partial t} + \frac{\partial y_m}{\partial t} + \sigma_{a_1} = 0 \\ \frac{\partial^2 y_m(t, \theta)}{\partial t^2} + a_2 \frac{\partial^2 \sigma_{a_2}}{\partial t^2} + a_1 \frac{\partial \sigma_{a_2}}{\partial t} + \sigma_{a_2} = 0 \\ a_2 \frac{\partial^2 \sigma_{b_0}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_0}}{\partial t} + \sigma_{b_0} = x(t) \\ a_2 \frac{\partial^2 \sigma_{b_1}}{\partial t^2} + a_1 \frac{\partial \sigma_{b_1}}{\partial t} + \sigma_{b_1} = \frac{dx(t)}{dt} \end{array} \right.$$

- Fonctions de transfert des fonctions de sensibilités :

$$\mathcal{L}(\sigma_{a_1}(t, \theta)) = \Sigma_{a_1}(p, \theta) = \left(\frac{-p}{a_2 p^2 + a_1 p + 1} \right) Y_m(p, \theta)$$

$$\mathcal{L}(\sigma_{a_2}(t, \theta)) = \Sigma_{a_2}(p, \theta) = \left(\frac{-p^2}{a_2 p^2 + a_1 p + 1} \right) Y_m(p, \theta)$$

$$\mathcal{L}(\sigma_{b_0}(t, \theta)) = \Sigma_{b_0}(p, \theta) = \left(\frac{1}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

$$\mathcal{L}(\sigma_{b_1}(t, \theta)) = \Sigma_{b_1}(p, \theta) = \left(\frac{p}{a_2 p^2 + a_1 p + 1} \right) X(p)$$

2.e Algorithme de Newton-Raphson

- **Initialisation**

Choisir θ , ε et $G > \varepsilon$

- **Tant que** ($\|G\| > \varepsilon$)

Définir a_1, \dots, b_1

Simuler $G_m(p) \rightarrow$ valeurs de $y_m(t)$

Simuler $H_{a_1}(p), \dots, H_{b_1}(p) \rightarrow$ valeurs de $\sigma_{a_1}(t), \dots, \sigma_{b_1}(t)$

Calculer G et H

Calculer $\theta \leftarrow \theta - H^{-1}G$

- **Fin**

Simuler $G_m(p) \rightarrow$ valeurs de $y_m(t)$

Tracer y_m et y

2.e Algorithme de Newton-Raphson

```
T= [1 2 3 4]';eps=0.001;G=1;dt=.25;
```

2.e Algorithme de Newton-Raphson

```
T= [1 2 3 4]';eps=0.001;G=1;dt=.25;  
while (abs(G)>eps)
```

```
end
```

2.e Algorithme de Newton-Raphson

```
T= [1 2 3 4]';eps=0.001;G=1;dt=.25;  
while (abs(G)>eps)  
    a1=T(1);a2=T(2);b0=T(3);b1=T(4);
```

```
end
```

2.e Algorithme de Newton-Raphson

```
T= [1 2 3 4]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a1=T(1);a2=T(2);b0=T(3);b1=T(4);
    Gm=tf([b1 b0],[a2 a1 1]);
    ym=lsim(Gm,x,t);
```

end

2.e Algorithm de Newton-Raphson

```
T= [1 2 3 4]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a1=T(1);a2=T(2);b0=T(3);b1=T(4);
    Gm=tf([b1 b0],[a2 a1 1]);
    ym=lsim(Gm,x,t);
    sa1=lsim(tf([-1 0],[a2 a1 1]),ym,t);
    sa2=...
    sb0=...
    sb1=...
```

end

2.e Algorithm de Newton-Raphson

```
T= [1 2 3 4]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a1=T(1);a2=T(2);b0=T(3);b1=T(4);
    Gm=tf([b1 b0],[a2 a1 1]);
    ym=lsim(Gm,x,t);
    sa1=lsim(tf([-1 0],[a2 a1 1]),ym,t);
    sa2=...
    sb0=...
    sb1=...
    G=[sa1 sa2 sb0 sb1]'*(ym-y)*dt;
    H=...
end
```

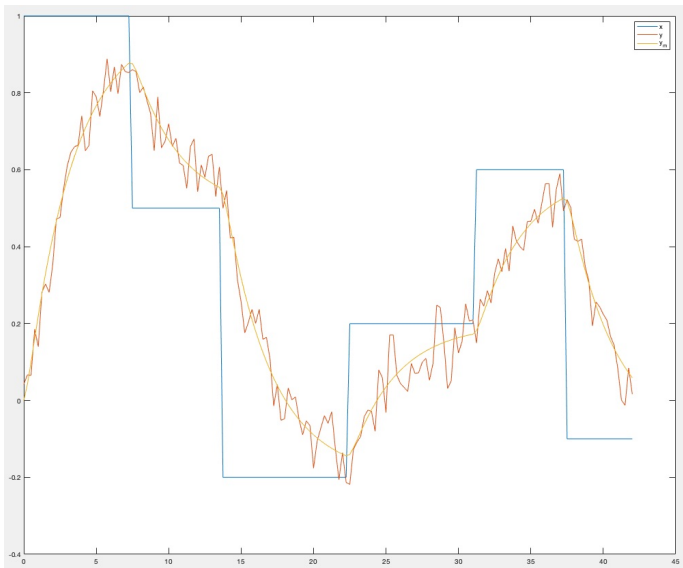
2.e Algorithm de Newton-Raphson

```
T= [1 2 3 4]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a1=T(1);a2=T(2);b0=T(3);b1=T(4);
    Gm=tf([b1 b0],[a2 a1 1]);
    ym=lsim(Gm,x,t);
    sa1=lsim(tf([-1 0],[a2 a1 1]),ym,t);
    sa2=...
    sb0=...
    sb1=...
    G=[sa1 sa2 sb0 sb1]'*(ym-y)*dt;
    H=...
    T=T-H^-1*G;
end
```

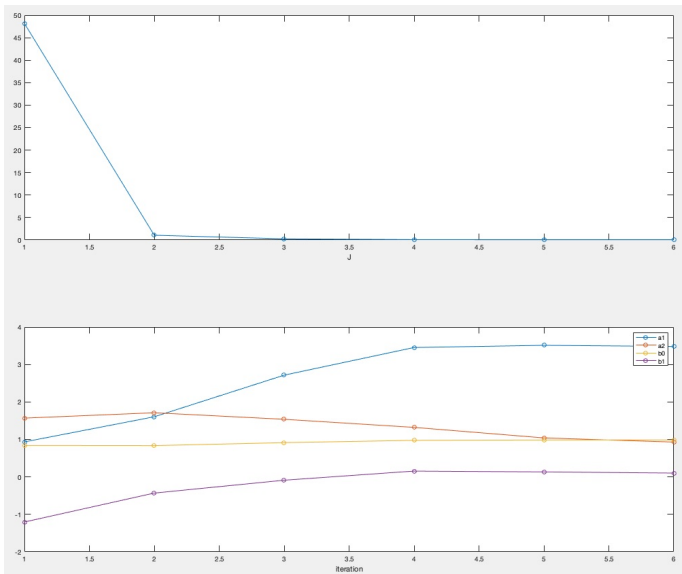

2.e Algorithm de Newton-Raphson

```
T= [1 2 3 4]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a1=T(1);a2=T(2);b0=T(3);b1=T(4);
    Gm=tf([b1 b0],[a2 a1 1]);
    ym=lsim(Gm,x,t);
    sa1=lsim(tf([-1 0],[a2 a1 1]),ym,t);
    sa2=...
    sb0=...
    sb1=...
    G=[sa1 sa2 sb0 sb1]'*(ym-y)*dt;
    H=...
    T=T-H^-1*G;
end
a1=T(1);a2=T(2);b0=T(3);b1=T(4);
Gm=tf([b1 b0],[a2 a1 1]);
ym=lsim(Gm,x,t);
plot(t,x,t,y,t,ym),legend('x','y','ym')
```

2.f Comparaison sortie / sortie modélisée

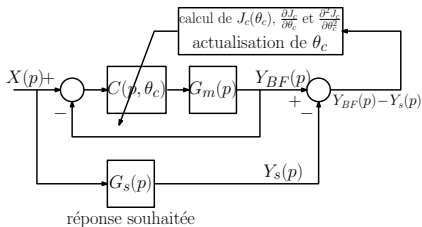


2.f Evolution du critère et des paramètres



3.a Synthèse du correcteur

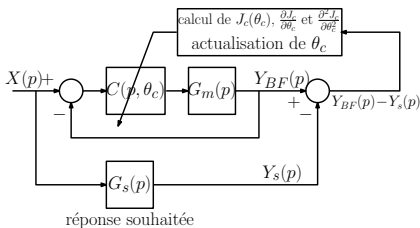
- On cherche un correcteur PID : $C(p) = \frac{a+bp+cp^2}{p}$



3.a Synthèse du correcteur

- On cherche un correcteur PID : $C(p) = \frac{a+bp+cp^2}{p}$
- Trouver $\theta_c = [a \ b \ c]^T$ qui minimise : $J_c(\theta_c) = \frac{1}{2} \int_0^\infty (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
où y_s est la sortie souhaitée en BF : $G_0 = 1$ et $t_p = 1.5$ s :

$$G_s(p) = \frac{1}{0.1357p^2 + 0.4687p + 1}$$



3.a Synthèse du correcteur

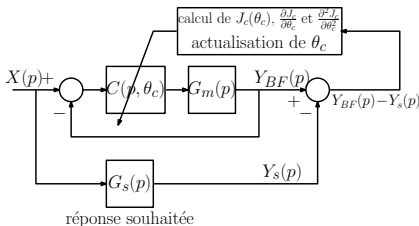
- On cherche un correcteur PID : $C(p) = \frac{a+bp+cp^2}{p}$
- Trouver $\theta_c = [a \ b \ c]^T$ qui minimise : $J_c(\theta_c) = \frac{1}{2} \int_0^\infty (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
où y_s est la sortie souhaitée en BF : $G_0 = 1$ et $t_p = 1.5$ s :

$$G_s(p) = \frac{1}{0.1357p^2 + 0.4687p + 1}$$

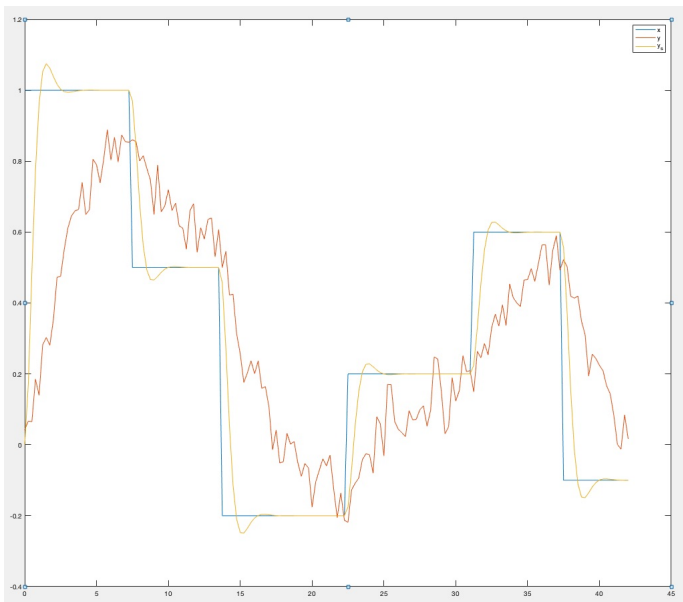
- Minimisation itérative par l'algorithme de **Newton-Raphson** :

$$\theta_{c,k+1} = \theta_{c,k} - (J_c''(\theta_{c,k}))^{-1} (J_c'(\theta_{c,k}))$$

- A chaque itération il faut calculer le **gradient** $G(\theta_{c,k}) = J_c'(\theta_{c,k})$ et le **hessien** $H(\theta_{c,k}) = J_c''(\theta_{c,k})$ du critère



3.b Comparaison sortie souhaitée en BF / sortie en BO



3.c Calculs des gradient et hessien

- Critère : $J_c(\theta_c) = \frac{1}{2} \int_0^{\infty} (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
- Gradient (dérivée première) :

3.c Calculs des gradient et hessien

- Critère : $J_c(\theta_c) = \frac{1}{2} \int_0^{\infty} (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
- Gradient (dérivée première) :

$$\frac{dJ_c(\theta_c)}{d\theta} = \int_0^{\infty} \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right) (y_{BF}(t, \theta) - y_s(t)) dt$$

3.c Calculs des gradient et hessien

- Critère : $J_c(\theta_c) = \frac{1}{2} \int_0^\infty (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
- Gradient (dérivée première) :

$$\frac{dJ_c(\theta_c)}{d\theta} = \int_0^\infty \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right) (y_{BF}(t, \theta) - y_s(t)) dt$$

Approché par une somme de Riemann :

$$\frac{dJ_c(\theta_c)}{d\theta_c} \approx \sum_{n=1}^N \left(\frac{\partial y_{BF}(t_n, \theta_c)}{\partial \theta_c} \right) (y_{BF}(t_n, \theta) - y_s(t_n)) \delta t$$

3.c Calculs des gradient et hessien

- Critère : $J_c(\theta_c) = \frac{1}{2} \int_0^\infty (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
- Gradient (dérivée première) :

$$\frac{dJ_c(\theta_c)}{d\theta} = \int_0^\infty \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right) (y_{BF}(t, \theta) - y_s(t)) dt$$

Approché par une somme de Riemann :

$$\frac{dJ_c(\theta_c)}{d\theta_c} \approx \sum_{n=1}^N \left(\frac{\partial y_{BF}(t_n, \theta_c)}{\partial \theta_c} \right) (y_{BF}(t_n, \theta) - y_s(t_n)) \delta t$$

Sous forme matricielle :

$$\frac{dJ_c(\theta_c)}{d\theta_c} \approx \begin{bmatrix} \sigma_a(t_1) & \sigma_a(t_2) & \dots & \sigma_a(t_N) \\ \sigma_b(t_1) & \sigma_b(t_2) & \dots & \sigma_b(t_N) \\ \sigma_c(t_1) & \sigma_c(t_2) & \dots & \sigma_c(t_N) \end{bmatrix} \begin{bmatrix} y_{BF}(t_1, \theta) - y_s(t_1) \\ y_{BF}(t_2, \theta) - y_s(t_2) \\ \vdots \\ y_{BF}(t_N, \theta) - y_s(t_N) \end{bmatrix} \delta t$$

3.c Calculs des gradient et hessien

- Critère : $J_c(\theta_c) = \frac{1}{2} \int_0^{\infty} (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
- Hessien (dérivée seconde) :

3.c Calculs des gradient et hessien

- Critère : $J_c(\theta_c) = \frac{1}{2} \int_0^\infty (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
- Hessien (dérivée seconde) :

$$\frac{d^2 J_c(\theta_c)}{d\theta_c^2} \approx \int_0^\infty \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right) \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right)^T dt$$

3.c Calculs des gradient et hessien

- Critère : $J_c(\theta_c) = \frac{1}{2} \int_0^\infty (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
- Hessien (dérivée seconde) :

$$\frac{d^2 J_c(\theta_c)}{d\theta_c^2} \approx \int_0^\infty \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right) \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right)^T dt$$

Approché par une somme de Riemann :

$$\frac{d^2 J_c(\theta_c)}{d\theta_c^2} \approx \sum_{n=1}^N \left(\frac{\partial y_{BF}(t_n, \theta_c)}{\partial \theta_c} \right) \left(\frac{\partial y_{BF}(t_n, \theta_c)}{\partial \theta_c} \right)^T \delta t$$

3.c Calculs des gradient et hessien

- Critère : $J_c(\theta_c) = \frac{1}{2} \int_0^\infty (y_{BF}(t, \theta_c) - y_s(t))^2 dt$
- Hessien (dérivée seconde) :

$$\frac{d^2 J_c(\theta_c)}{d\theta_c^2} \approx \int_0^\infty \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right) \left(\frac{\partial y_{BF}(t, \theta_c)}{\partial \theta_c} \right)^T dt$$

Approché par une somme de Riemann :

$$\frac{d^2 J_c(\theta_c)}{d\theta_c^2} \approx \sum_{n=1}^N \left(\frac{\partial y_{BF}(t_n, \theta_c)}{\partial \theta_c} \right) \left(\frac{\partial y_{BF}(t_n, \theta_c)}{\partial \theta_c} \right)^T \delta t$$

Sous forme matricielle :

$$\frac{d^2 J_c(\theta_c)}{d\theta_c^2} \approx \begin{bmatrix} \sigma_a(t_1) & \sigma_a(t_2) & \dots & \sigma_a(t_N) \\ \sigma_b(t_1) & \sigma_b(t_2) & \dots & \sigma_b(t_N) \\ \sigma_c(t_1) & \sigma_c(t_2) & \dots & \sigma_c(t_N) \end{bmatrix} \begin{bmatrix} \sigma_a(t_1) & \sigma_b(t_1) & \sigma_c(t_1) \\ \sigma_a(t_2) & \sigma_b(t_2) & \sigma_c(t_2) \\ \vdots & \vdots & \vdots \\ \sigma_a(t_N) & \sigma_b(t_N) & \sigma_c(t_N) \end{bmatrix} \delta t$$

3.d Fonctions de sensibilité

- Relation entrée/sortie du système en BF $\left(H_{BF} = \frac{CG}{1+CG}\right)$:

3.d Fonctions de sensibilité

- Relation entrée/sortie du système en BF ($H_{BF} = \frac{CG}{1+CG}$) :

$$Y_{BF}(p, \theta_c) = \frac{ab_0 + (ab_1 + bb_0)p + (cb_0 + bb_1)p^2 + cb_1p^3}{ab_0 + (ab_1 + bb_0 + 1)p + (cb_0 + bb_1 + a_1)p^2 + (cb_1 + a_2)p^3} X(p)$$

3.d Fonctions de sensibilité

- Relation entrée/sortie du système en BF ($H_{BF} = \frac{CG}{1+CG}$) :

$$Y_{BF}(p, \theta_c) = \frac{ab_0 + (ab_1 + bb_0)p + (cb_0 + bb_1)p^2 + cb_1p^3}{ab_0 + (ab_1 + bb_0 + 1)p + (cb_0 + bb_1 + a_1)p^2 + (cb_1 + a_2)p^3} X(p)$$

- Equation différentielle équivalente :

3.d Fonctions de sensibilité

- Relation entrée/sortie du système en BF ($H_{BF} = \frac{CG}{1+CG}$) :

$$Y_{BF}(p, \theta_c) = \frac{ab_0 + (ab_1 + bb_0)p + (cb_0 + bb_1)p^2 + cb_1p^3}{ab_0 + (ab_1 + bb_0 + 1)p + (cb_0 + bb_1 + a_1)p^2 + (cb_1 + a_2)p^3} X(p)$$

- Equation différentielle équivalente :

$$ab_0 y_{BF} + (ab_1 + bb_0 + 1)y'_{BF} + (cb_0 + bb_1 + a_1)y''_{BF} + (cb_1 + a_2)y'''_{BF} = ab_0 x + (ab_1 + bb_0)x' + (cb_0 + bb_1)x'' + cb_1 x'''$$

3.d Fonctions de sensibilité

- Relation entrée/sortie du système en BF ($H_{BF} = \frac{CG}{1+CG}$) :

$$Y_{BF}(p, \theta_c) = \frac{ab_0 + (ab_1 + bb_0)p + (cb_0 + bb_1)p^2 + cb_1p^3}{ab_0 + (ab_1 + bb_0 + 1)p + (cb_0 + bb_1 + a_1)p^2 + (cb_1 + a_2)p^3} X(p)$$

- Equation différentielle équivalente :

$$ab_0 y_{BF} + (ab_1 + bb_0 + 1)y'_{BF} + (cb_0 + bb_1 + a_1)y''_{BF} + (cb_1 + a_2)y'''_{BF} = ab_0 x + (ab_1 + bb_0)x' + (cb_0 + bb_1)x'' + cb_1 x'''$$

- Dérivées par rapport à a , b et c :

3.d Fonctions de sensibilité

- Relation entrée/sortie du système en BF ($H_{BF} = \frac{CG}{1+CG}$) :

$$Y_{BF}(p, \theta_c) = \frac{ab_0 + (ab_1 + bb_0)p + (cb_0 + bb_1)p^2 + cb_1p^3}{ab_0 + (ab_1 + bb_0 + 1)p + (cb_0 + bb_1 + a_1)p^2 + (cb_1 + a_2)p^3} X(p)$$

- Equation différentielle équivalente :

$$ab_0 y_{BF} + (ab_1 + bb_0 + 1)y'_{BF} + (cb_0 + bb_1 + a_1)y''_{BF} + (cb_1 + a_2)y'''_{BF} = ab_0 x + (ab_1 + bb_0)x' + (cb_0 + bb_1)x'' + cb_1 x'''$$

- Dérivées par rapport à a , b et c :

$$ab_0 \sigma_a + (ab_1 + bb_0 + 1)\sigma'_a + (cb_0 + bb_1 + a_1)\sigma''_a + (cb_1 + a_2)\sigma'''_a = b_0(x - y_{BF}) + b_1(x' - y'_{BF})$$

3.d Fonctions de sensibilité

- Relation entrée/sortie du système en BF ($H_{BF} = \frac{CG}{1+CG}$) :

$$Y_{BF}(p, \theta_c) = \frac{ab_0 + (ab_1 + bb_0)p + (cb_0 + bb_1)p^2 + cb_1p^3}{ab_0 + (ab_1 + bb_0 + 1)p + (cb_0 + bb_1 + a_1)p^2 + (cb_1 + a_2)p^3} X(p)$$

- Equation différentielle équivalente :

$$ab_0 y_{BF} + (ab_1 + bb_0 + 1)y'_{BF} + (cb_0 + bb_1 + a_1)y''_{BF} + (cb_1 + a_2)y'''_{BF} = ab_0 x + (ab_1 + bb_0)x' + (cb_0 + bb_1)x'' + cb_1 x'''$$

- Dérivées par rapport à a , b et c :

$$ab_0 \sigma_a + (ab_1 + bb_0 + 1)\sigma'_a + (cb_0 + bb_1 + a_1)\sigma''_a + (cb_1 + a_2)\sigma'''_a = b_0(x - y_{BF}) + b_1(x' - y'_{BF})$$

$$ab_0 \sigma_b + (ab_1 + bb_0 + 1)\sigma'_b + (cb_0 + bb_1 + a_1)\sigma''_b + (cb_1 + a_2)\sigma'''_b = b_0(x' - y'_{BF}) + b_1(x'' - y''_{BF})$$

3.d Fonctions de sensibilité

- Relation entrée/sortie du système en BF ($H_{BF} = \frac{CG}{1+CG}$) :

$$Y_{BF}(p, \theta_c) = \frac{ab_0 + (ab_1 + bb_0)p + (cb_0 + bb_1)p^2 + cb_1p^3}{ab_0 + (ab_1 + bb_0 + 1)p + (cb_0 + bb_1 + a_1)p^2 + (cb_1 + a_2)p^3} X(p)$$

- Equation différentielle équivalente :

$$ab_0 y_{BF} + (ab_1 + bb_0 + 1)y'_{BF} + (cb_0 + bb_1 + a_1)y''_{BF} + (cb_1 + a_2)y'''_{BF} = ab_0 x + (ab_1 + bb_0)x' + (cb_0 + bb_1)x'' + cb_1 x'''$$

- Dérivées par rapport à a , b et c :

$$ab_0 \sigma_a + (ab_1 + bb_0 + 1)\sigma'_a + (cb_0 + bb_1 + a_1)\sigma''_a + (cb_1 + a_2)\sigma'''_a = b_0(x - y_{BF}) + b_1(x' - y'_{BF})$$

$$ab_0 \sigma_b + (ab_1 + bb_0 + 1)\sigma'_b + (cb_0 + bb_1 + a_1)\sigma''_b + (cb_1 + a_2)\sigma'''_b = b_0(x' - y'_{BF}) + b_1(x'' - y''_{BF})$$

$$ab_0 \sigma_c + (ab_1 + bb_0 + 1)\sigma'_c + (cb_0 + bb_1 + a_1)\sigma''_c + (cb_1 + a_2)\sigma'''_c = b_0(x'' - y''_{BF}) + b_1(x''' - y'''_{BF})$$

3.e Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\begin{cases} ab_0\sigma_a+(ab_1+bb_0+1)\sigma'_a+(cb_0+bb_1+a_1)\sigma''_a+(cb_1+a_2)\sigma'''_a=b_0(x-y_{BF})+b_1(x'-y'_{BF}) \\ ab_0\sigma_b+(ab_1+bb_0+1)\sigma'_b+(cb_0+bb_1+a_1)\sigma''_b+(cb_1+a_2)\sigma'''_b=b_0(x'-y'_{BF})+b_1(x''-y''_{BF}) \\ ab_0\sigma_c+(ab_1+bb_0+1)\sigma'_c+(cb_0+bb_1+a_1)\sigma''_c+(cb_1+a_2)\sigma'''_c=b_0(x''-y''_{BF})+b_1(x'''-y'''_{BF}) \end{cases}$$

3.e Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\begin{cases} ab_0\sigma_a+(ab_1+bb_0+1)\sigma'_a+(cb_0+bb_1+a_1)\sigma''_a+(cb_1+a_2)\sigma'''_a = b_0(x-y_{BF})+b_1(x'-y'_{BF}) \\ ab_0\sigma_b+(ab_1+bb_0+1)\sigma'_b+(cb_0+bb_1+a_1)\sigma''_b+(cb_1+a_2)\sigma'''_b = b_0(x'-y'_{BF})+b_1(x''-y''_{BF}) \\ ab_0\sigma_c+(ab_1+bb_0+1)\sigma'_c+(cb_0+bb_1+a_1)\sigma''_c+(cb_1+a_2)\sigma'''_c = b_0(x''-y''_{BF})+b_1(x'''-y'''_{BF}) \end{cases}$$

- Fonctions de transfert des fonctions de sensibilités :

3.e Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\begin{cases} ab_0\sigma_a + (ab_1 + bb_0 + 1)\sigma'_a + (cb_0 + bb_1 + a_1)\sigma''_a + (cb_1 + a_2)\sigma'''_a = b_0(x - y_{BF}) + b_1(x' - y'_{BF}) \\ ab_0\sigma_b + (ab_1 + bb_0 + 1)\sigma'_b + (cb_0 + bb_1 + a_1)\sigma''_b + (cb_1 + a_2)\sigma'''_b = b_0(x' - y'_{BF}) + b_1(x'' - y''_{BF}) \\ ab_0\sigma_c + (ab_1 + bb_0 + 1)\sigma'_c + (cb_0 + bb_1 + a_1)\sigma''_c + (cb_1 + a_2)\sigma'''_c = b_0(x'' - y''_{BF}) + b_1(x''' - y'''_{BF}) \end{cases}$$

- Fonctions de transfert des fonctions de sensibilités :

$$\mathcal{L}(\sigma_a(t, \theta_c)) = \Sigma_a(p, \theta_c) = \left(\frac{b_0 + b_1 p}{ab_0 + \dots + (cb_1 + a_2)p^3} \right) (X(p) - Y_{BF}(p, \theta_c))$$

3.e Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\begin{cases} ab_0\sigma_a + (ab_1 + bb_0 + 1)\sigma'_a + (cb_0 + bb_1 + a_1)\sigma''_a + (cb_1 + a_2)\sigma'''_a = b_0(x - y_{BF}) + b_1(x' - y'_{BF}) \\ ab_0\sigma_b + (ab_1 + bb_0 + 1)\sigma'_b + (cb_0 + bb_1 + a_1)\sigma''_b + (cb_1 + a_2)\sigma'''_b = b_0(x' - y'_{BF}) + b_1(x'' - y''_{BF}) \\ ab_0\sigma_c + (ab_1 + bb_0 + 1)\sigma'_c + (cb_0 + bb_1 + a_1)\sigma''_c + (cb_1 + a_2)\sigma'''_c = b_0(x'' - y''_{BF}) + b_1(x''' - y'''_{BF}) \end{cases}$$

- Fonctions de transfert des fonctions de sensibilités :

$$\mathcal{L}(\sigma_a(t, \theta_c)) = \Sigma_a(p, \theta_c) = \left(\frac{b_0 + b_1 p}{ab_0 + \dots + (cb_1 + a_2)p^3} \right) (X(p) - Y_{BF}(p, \theta_c))$$

$$\mathcal{L}(\sigma_b(t, \theta_c)) = \Sigma_b(p, \theta_c) = \left(\frac{b_0 p + b_1 p^2}{ab_0 + \dots + (cb_1 + a_2)p^3} \right) (X(p) - Y_{BF}(p, \theta_c))$$

3.e Fonctions de transfert des fonctions de sensibilité

- Equations différentielles vérifiées par les fonctions de sensibilité :

$$\begin{cases} ab_0\sigma_a + (ab_1 + bb_0 + 1)\sigma'_a + (cb_0 + bb_1 + a_1)\sigma''_a + (cb_1 + a_2)\sigma'''_a = b_0(x - y_{BF}) + b_1(x' - y'_{BF}) \\ ab_0\sigma_b + (ab_1 + bb_0 + 1)\sigma'_b + (cb_0 + bb_1 + a_1)\sigma''_b + (cb_1 + a_2)\sigma'''_b = b_0(x' - y'_{BF}) + b_1(x'' - y''_{BF}) \\ ab_0\sigma_c + (ab_1 + bb_0 + 1)\sigma'_c + (cb_0 + bb_1 + a_1)\sigma''_c + (cb_1 + a_2)\sigma'''_c = b_0(x'' - y''_{BF}) + b_1(x''' - y'''_{BF}) \end{cases}$$

- Fonctions de transfert des fonctions de sensibilités :

$$\mathcal{L}(\sigma_a(t, \theta_c)) = \Sigma_a(p, \theta_c) = \left(\frac{b_0 + b_1 p}{ab_0 + \dots + (cb_1 + a_2)p^3} \right) (X(p) - Y_{BF}(p, \theta_c))$$

$$\mathcal{L}(\sigma_b(t, \theta_c)) = \Sigma_b(p, \theta_c) = \left(\frac{b_0 p + b_1 p^2}{ab_0 + \dots + (cb_1 + a_2)p^3} \right) (X(p) - Y_{BF}(p, \theta_c))$$

$$\mathcal{L}(\sigma_c(t, \theta_c)) = \Sigma_c(p, \theta_c) = \left(\frac{b_0 p^2 + b_1 p^3}{ab_0 + \dots + (cb_1 + a_2)p^3} \right) (X(p) - Y_{BF}(p, \theta_c))$$

3.f Algorithme de Newton-Raphson

- **Simulation de $y_s(t)$**

Simuler $G_s(p) \rightarrow$ valeurs de $y_s(t)$

- **Initialisation**

Choisir θ_c , ε et $G > \varepsilon$

- **Tant que** ($\|G\| > \varepsilon$)

Définir a, b, c

Simuler $G_{BF}(p) \rightarrow$ valeurs de $y_{BF}(t)$

Simuler $H_a(p), H_b(p), H_c(p) \rightarrow$ valeurs de $\sigma_a(t), \sigma_b(t), \sigma_c(t)$

Calculer G et H

Calculer $\theta_c \leftarrow \theta_c - H^{-1}G$

- **Fin**

Simuler $G_{BF}(p) \rightarrow$ valeurs de $y_{BF}(t)$

Tracer y_s et y_{BF}

3.f Algorithme de Newton-Raphson

```
Gs=tf(1,[0.1357 0.4687 1]);  
ys=lsim(Gs,x,t);  
Tc= [1 2 3]';eps=0.001;G=1;dt=.25;
```

3.f Algorithme de Newton-Raphson

```
Gs=tf(1,[0.1357 0.4687 1]);  
ys=lsim(Gs,x,t);  
Tc= [1 2 3]';eps=0.001;G=1;dt=.25;  
while (abs(G)>eps)
```

```
end
```

3.f Algorithme de Newton-Raphson

```
Gs=tf(1,[0.1357 0.4687 1]);  
ys=lsim(Gs,x,t);  
Tc= [1 2 3]';eps=0.001;G=1;dt=.25;  
while (abs(G)>eps)  
    a=Tc(1);b=Tc(2);c=Tc(3);  
  
end
```


3.f Algorithme de Newton-Raphson

```
Gs=tf(1,[0.1357 0.4687 1]);
ys=lsim(Gs,x,t);
Tc= [1 2 3]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a=Tc(1);b=Tc(2);c=Tc(3);
    Den=[c*b1+a2  c*b0+b*b1+a1  a*b1+b*b0+1  a*b0]
    GBF=tf([c*b1  c*b0+b*b1  a*b1+b*b0  a*b0],Den);
    yBF=lsim(GBF,x,t);
```

end

3.f Algorithme de Newton-Raphson

```
Gs=tf(1,[0.1357 0.4687 1]);
ys=lsim(Gs,x,t);
Tc= [1 2 3]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a=Tc(1);b=Tc(2);c=Tc(3);
    Den=[c*b1+a2  c*b0+b*b1+a1  a*b1+b*b0+1  a*b0]
    GBF=tf([c*b1  c*b0+b*b1  a*b1+b*b0  a*b0],Den);
    yBF=lsim(GBF,x,t);
    sa=lsim(tf([b1 b0],Den),x-yBF,t);
    sb=...
    sc=...

end
```

3.f Algorithme de Newton-Raphson

```
Gs=tf(1,[0.1357 0.4687 1]);
ys=lsim(Gs,x,t);
Tc= [1 2 3]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a=Tc(1);b=Tc(2);c=Tc(3);
    Den=[c*b1+a2  c*b0+b*b1+a1  a*b1+b*b0+1  a*b0]
    GBF=tf([c*b1  c*b0+b*b1  a*b1+b*b0  a*b0],Den);
    yBF=lsim(GBF,x,t);
    sa=lsim(tf([b1 b0],Den),x-yBF,t);
    sb=...
    sc=...
    G=[sa sb sc]’*(yBF-ys)*dt;
    H=...
end
```

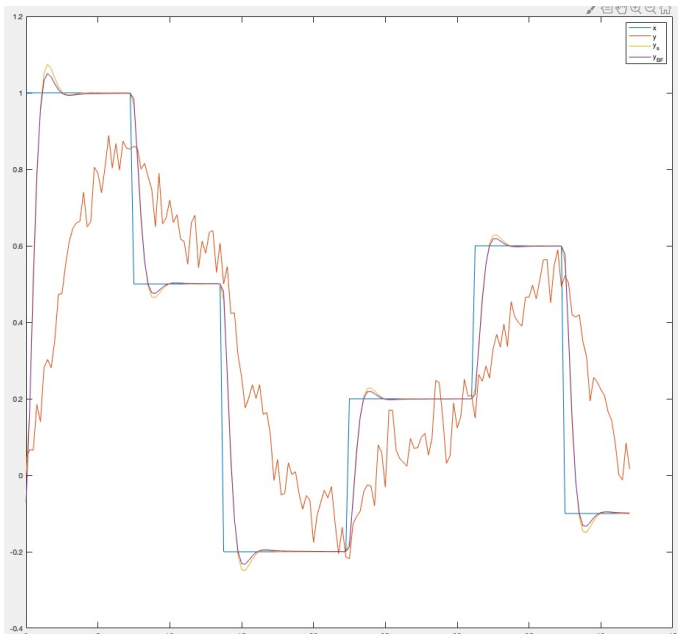
3.f Algorithme de Newton-Raphson

```
Gs=tf(1,[0.1357 0.4687 1]);
ys=lsim(Gs,x,t);
Tc= [1 2 3]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a=Tc(1);b=Tc(2);c=Tc(3);
    Den=[c*b1+a2  c*b0+b*b1+a1  a*b1+b*b0+1  a*b0]
    GBF=tf([c*b1  c*b0+b*b1  a*b1+b*b0  a*b0],Den);
    yBF=lsim(GBF,x,t);
    sa=lsim(tf([b1 b0],Den),x-yBF,t);
    sb=...
    sc=...
    G=[sa sb sc]’*(yBF-ys)*dt;
    H=...
    Tc=Tc-H^-1*G;
end
```

3.f Algorithme de Newton-Raphson

```
Gs=tf(1,[0.1357 0.4687 1]);
ys=lsim(Gs,x,t);
Tc= [1 2 3]';eps=0.001;G=1;dt=.25;
while (abs(G)>eps)
    a=Tc(1);b=Tc(2);c=Tc(3);
    Den=[c*b1+a2  c*b0+b*b1+a1  a*b1+b*b0+1  a*b0]
    GBF=tf([c*b1  c*b0+b*b1  a*b1+b*b0  a*b0],Den);
    yBF=lsim(GBF,x,t);
    sa=lsim(tf([b1 b0],Den),x-yBF,t);
    sb=...
    sc=...
    G=[sa sb sc]’*(yBF-ys)*dt;
    H=...
    Tc=Tc-H^-1*G;
end
a=Tc(1);b=Tc(2);c=Tc(3);
Den=[c*b1+a2  c*b0+b*b1+a1  a*b1+b*b0+1  a*b0]
GBF=tf([c*b1  c*b0+b*b1  a*b1+b*b0  a*b0], Den);
yBF=lsim(GBF,x,t);
plot(t,x,t,ys,t,yBF),legend('x','ys','yBF')
```

3.g Comparaison sortie BF / sortie souhaitée



3.g Evolution du critère et des paramètres

