

Chapitre 6

EXEMPLES DE COMMANDE DE PROCESSUS

INTRODUCTION.

Le but de ce chapitre est d'illustrer la mise en œuvre et les propriétés des divers systèmes de commande présentés au chapitre 5 précédent. Le problème de la régulation de l'état d'un processus étant bien illustré par la commande latérale du véhicule REMI, nous ne l'abordons pas ici. Nous portons nos efforts sur la comparaison des systèmes de commande de poursuite et de régulation de sortie, et nous illustrons les propriétés de ces systèmes avec un processus non linéaire. Ce dernier est le processus simulé par le modèle d'état identifié au chapitre 4 §II.3.2.2.

Par ailleurs, les méthodes du chapitre 5 n'étant pas optimales par rapport au bruit, nous montrons rapidement au §II qu'il est néanmoins possible de construire des systèmes de commande à variance minimale en utilisant le correcteur-S, dans les cas NARX et NBSX. Ceci est illustré à l'aide du processus simulé par le modèle entrée-sortie identifié au chapitre 4 §I.3.2.1.

I. POURSUITE ET RÉGULATION DE LA SORTIE.

I.1. PRÉSENTATION.

Nous étudions les performances des correcteurs -S et -D au sein de systèmes de commande par simple bouclage (SCSB), et avec modèle interne (SCMI). Notre but est ici de mettre en évidence les propriétés de robustesse de ces systèmes de commande par rapport à des perturbations déterministes non mesurées, ainsi que par rapport à des défauts de modélisation ou d'apprentissage des correcteurs.

Processus.

Le processus simulé est le suivant :

$$\begin{cases} x_1(k+1) = a_1 x_1(k) + a_2 x_2(k) + (b_1 + 2 b_2) u(k) \\ x_2(k+1) = \frac{x_1(k)}{1 + 0,01 x_2(k)^2} + \frac{(- b_2)}{a_2} u(k) \\ y(k) = 4 \tanh\left(\frac{x_1(k)}{4}\right) \end{cases}$$

avec $a_1 = 1,145$; $a_2 = -0,549$; $b_1 = 0,222$; $b_2 = 0,181$. Ces coefficients sont obtenus à partir de la discrétisation du filtre linéaire du second ordre de pulsation $\omega_n=3$, d'amortissement $\xi=0,4$ et de gain 1 (voir chapitre 4 §II.1). Le modèle est stable et à inverse stable dans le domaine de fonctionnement décrit par le processus lors de l'identification (la commande varie entre ± 5). L'apprentissage des

correcteurs -S et -D, et leur utilisation au sein de SCSB et SCMI ne pose donc pas de problème théorique.

Objectifs de commande.

La dynamique de poursuite désirée pour le système de commande, ou dynamique de référence, est définie par le modèle obtenu en discrétisant le second ordre linéaire de pulsation $\omega_n=3$, d'amortissement $\xi=0,7$ et de gain 1, que nous notons :

$$E(q) y_r(k+1) = H(q) r(k)$$

avec :

$$E(q) = 1 + e_1 q^{-1} + e_2 q^{-2} = 1 - 1,017 q^{-1} + 0,350 q^{-2} ; H(q) = h_1 q^{-1} + h_2 q^{-2} = 0,195 q^{-1} + 0,137 q^{-2}.$$

Ce modèle est plus rapide que le processus non asservi ; il est beaucoup plus amorti que le processus pour les petites amplitudes ; son gain statique est constant, toujours supérieur à celui du processus, qui lui décroît avec l'amplitude de la commande (voir chapitre 4 §II.1). D'autre part, nous nous intéressons à la dynamique de régulation par rapport à des perturbations additives de sortie, et à la compensation de ces perturbations, comme cela a été fait au chapitre 5 §II, et dans l'annexe II pour des systèmes linéaires.

I.2. APPRENTISSAGE DES CORRECTEURS -S ET -D.

Modèle de référence d'apprentissage.

Sa sortie y_a obéit à : $E_a(q) y_a(k+1) = H_a(q) r(k)$, avec (cf. chapitre 5 §II.2.1) :

- dans le cas d'un correcteur-S, $E_a(q) = 1 ; H_a(q) = 1$,
- dans le cas d'un correcteur-D, $E_a(q) = E(q) ; H_a(q) = H(q)$.

Modèle de simulation.

Nous avons conservé le modèle de simulation neuronal établi à partir du prédicteur entrée-sortie non bouclé à cinq neurones cachés identifié au chapitre 4 §II.3.2.2 :

$$y(k) = \varphi_{RN}^{Sim2}(y(k-1), y(k-2), u(k-1), u(k-2))$$

où φ_{RN}^{Sim2} est la fonction réalisée par la partie non bouclée du réseau obtenu en fin d'identification.

Correcteurs -S et -D.

Ces correcteurs sont bouclés. Il faut pour les réaliser un réseau de neurones bouclé de la forme (cf. chapitre 5 §II.2.1) :

$$u(k) = \varphi_{RN}(H_a(q) r(k), y(k), y(k-1), u(k-1)); C)$$

Nous utilisons un réseau de même complexité que le modèle, c'est-à-dire possédant 5 neurones cachés complètement connectés. La fonction d'activation du neurone de sortie est une sigmoïde d'amplitude 5, ainsi choisie pour ne pas sortir du domaine de fonctionnement dans lequel le modèle de simulation est valable.

Algorithme d'apprentissage.

Nous choisissons de réaliser un apprentissage semi-dirigé, où ni les entrées d'état du modèle, ni celles du correcteur, ne sont dirigées par les sorties du modèle de référence (l'algorithme est nécessairement semi-dirigé puisque le correcteur est bouclé).

Le système d'apprentissage complet est représenté sur la figure 1.

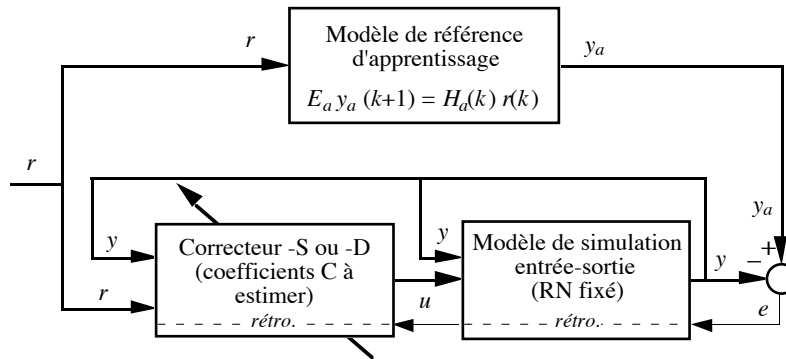


Figure 1.
Système d'apprentissage des correcteurs -S et -D en semi-dirigé.

Séquences d'apprentissage.

La sortie du modèle se stabilise à environ $\pm 3,5$ pour la commande d'amplitude maximale ± 5 . Afin que les performances respectives des correcteurs -S et -D soient comparables, nous avons choisi dans chaque cas :

- correcteur-S : la séquence de consigne d'apprentissage $\{r(k)\}$ est calculée à l'aide du modèle de référence définissant la dynamique de poursuite désirée (polynômes E et H), à partir d'une suite de créneaux d'amplitudes aléatoires dans l'intervalle $[-3,5 ; 3,5]$, et de durée 10 pas d'échantillonnage. La séquence totale comporte 1000 pas d'échantillonnage. La séquence de référence d'apprentissage $\{y_a(k)\}$, avec $y_a(k+1) = r(k)$, est représentée sur la figure 2 suivante.
- correcteur-D : la séquence de consigne d'apprentissage $\{r(k)\}$ est la séquence de créneaux précédente. $\{y_a(k)\}$ est calculée à l'aide du modèle de référence d'apprentissage ($E_a(q) = E(q) ; H_a(q) = H(q)$), et coïncide avec la séquence de référence d'apprentissage du correcteur-S. Ces séquences sont représentées sur la figure 3.

Séquences d'estimation de la performance.

Nous présentons les résultats obtenus en fin d'apprentissage sur une séquence de test destinée à discriminer les différents systèmes de commande, en poursuite et en particulier en régulation. Il s'agit d'une série de 5 paliers de consigne de durée 50 pas d'échantillonnage. Une perturbation déterministe additive en sortie d'amplitude $+0,5$ est appliquée entre les instants 110 et 140 (voir les figures 6a et 6b par exemple). La perturbation de sortie y est signalée par une barre grisée sur l'axe des abscisses.

I.2.1. Correcteur-S.

La performance du correcteur-S en fin d'apprentissage est représentée sur la figure 2.

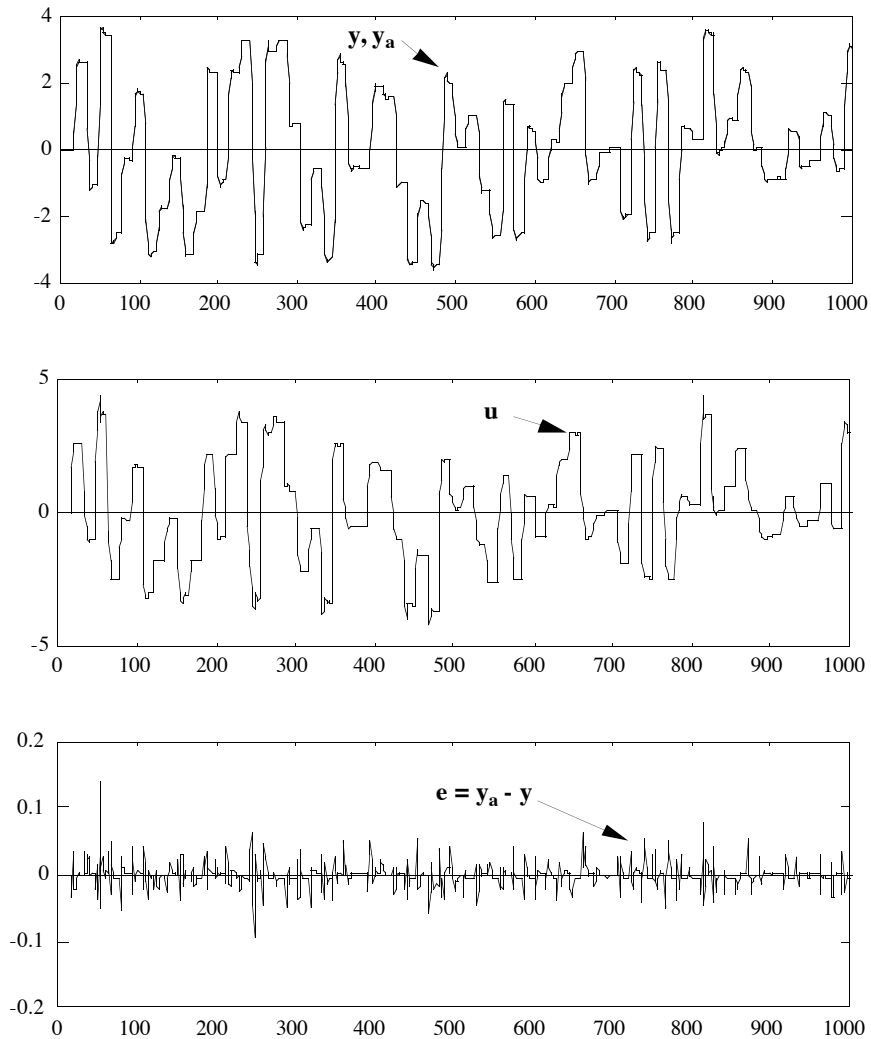


Figure 2.
Performance du correcteur-S sur la séquence d'apprentissage.

Ces résultats montrent que le correcteur obtenu est assez précis pour être utilisé dans un SCMI : on n'observe de grandes erreurs que pour des amplitudes ou des variations de consigne importantes, c'est-à-dire lorsque l'inverse théorique du modèle n'est plus réalisable à cause de la saturation en ± 5 du correcteur. Le comportement général de la commande est bien celui attendu : amortir le processus aux petites amplitudes, où il est oscillant, et augmenter son gain aux grandes. Nous notons :

$$u(k) = \varphi_{RN}^S(r(k), y(k), y(k-1), u(k-1))$$

le correcteur-S obtenu en fin d'apprentissage

I.2.2. Correcteur-D.

Les résultats obtenus avec le correcteur-D en fin d'apprentissage sont représentés sur la figure 3. La qualité de l'apprentissage est comparable à celle du correcteur-S, et les remarques générales sont les mêmes que pour ce dernier. Nous notons :

$$u(k) = \varphi_{RN}^D(H(q) r(k), y(k), y(k-1), u(k-1))$$

le correcteur-D obtenu en fin d'apprentissage.

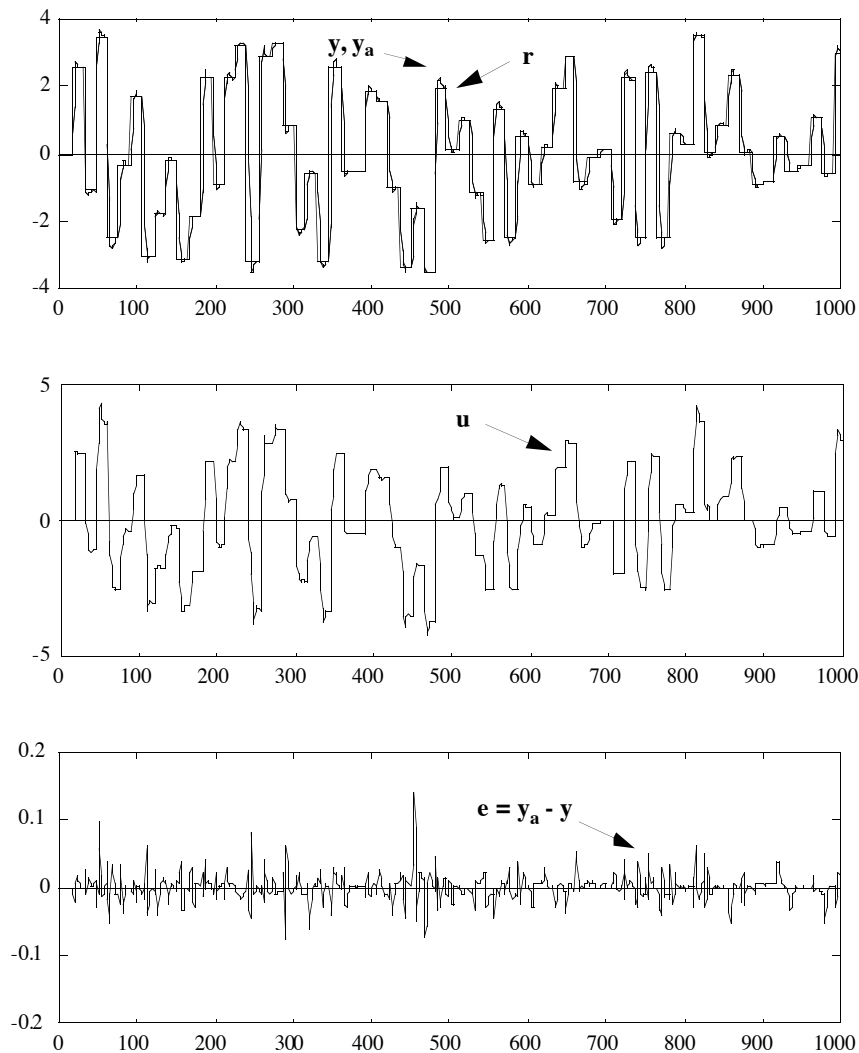


Figure 3.
Performance du correcteur-D sur la séquence d'apprentissage.

I.3. TEST DES CORRECTEURS AVEC LE MODÈLE PERTURBÉ.

Afin d'observer le comportement en régulation du système nominal en réponse à une perturbation de sortie en créneau, nous soumettons les deux correcteurs en simple bouclage avec le modèle (comme pendant l'apprentissage) à la séquence de test.

I.3.1. Correcteur-S.

La figure 4 représente le schéma bloc du système. Le correcteur-S est utilisé avec les arguments :

$$u(k) = \varphi_{RN}^S(y_r(k+1), y(k), y(k-1), u(k-1))$$

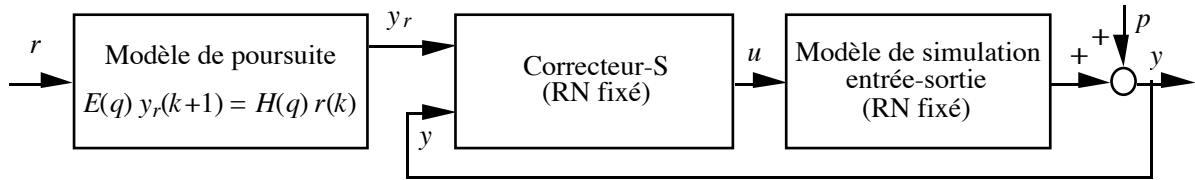


Figure 4.
Test avec le modèle perturbé du correcteur-S.

Les résultats correspondants sont rassemblés sur la figure 5a. Comme on pouvait le prévoir d'après l'annexe II §II.1.1.1 et le chapitre 5 §II.3.1.1, la perturbation provoque une erreur statique non négligeable, et de brusques variations de la commande, de faible amplitude toutefois.

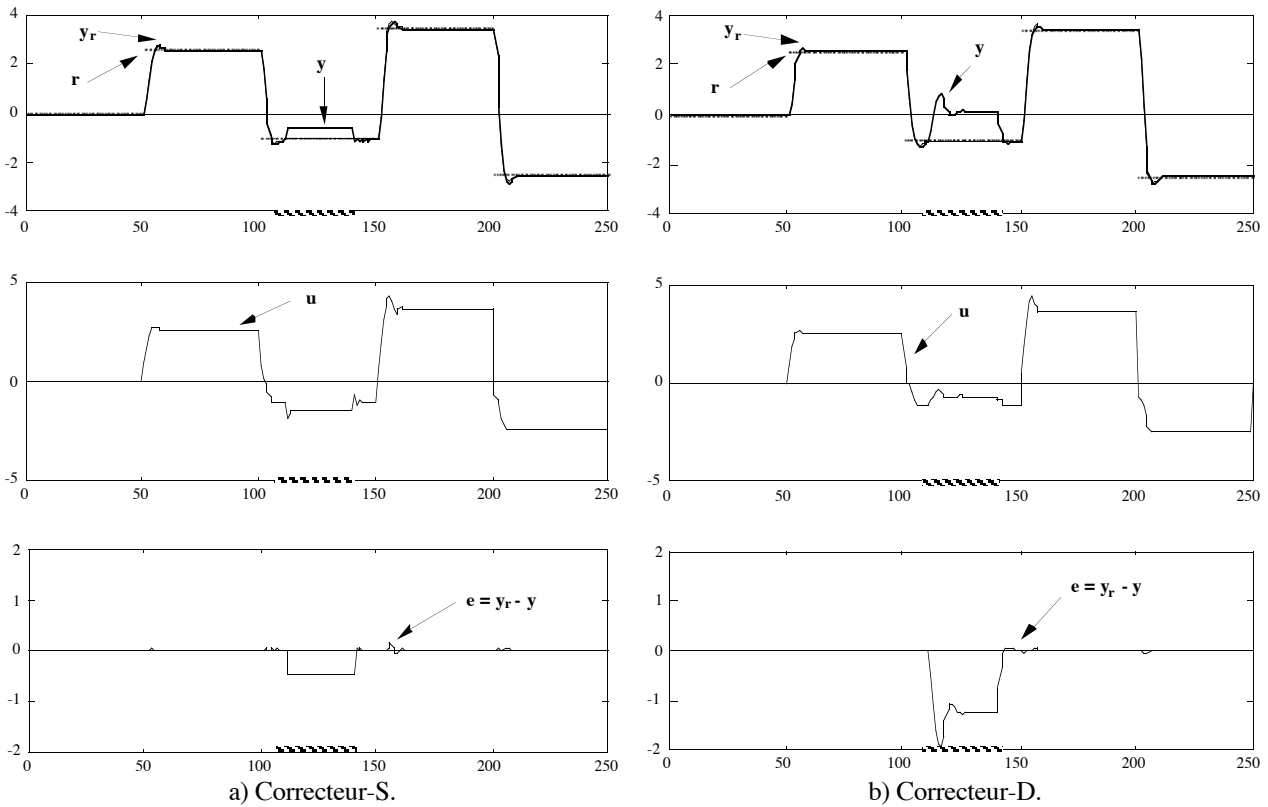


Figure 5.
Performances sur la séquence de test avec le modèle perturbé.

I.3.2. Correcteur-D.

La figure 6 représente le schéma bloc du système. Le correcteur est utilisé avec les arguments :

$$u(k) = \varphi_{RN}^D(H(q) r(k), y(k), y(k-1), u(k-1))$$

La séquence $\{y_r(k)\}$, qui sert à évaluer la performance des systèmes de commande, est calculée, comme dans le cas précédent, par le modèle de référence donnant la dynamique désirée, c'est-à-dire $E(q) y_r(k+1) = H(q) r(k)$.

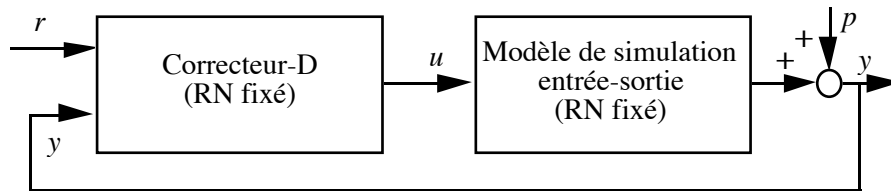


Figure 6.
Test avec le modèle perturbé du correcteur-D.

Les résultats correspondant sont rassemblés sur la figure 5b. Comme il était prévisible (annexe II §II.1.2 et chapitre 5 §II.3.2.1), l'erreur statique provoquée par la perturbation de sortie est plus importante que pour le correcteur-S avec un modèle de poursuite, mais la dynamique de régulation est moins brusque, et ne provoque pas d'à-coups sur la commande : la dynamique de régulation est la même que la dynamique de poursuite.

I.4. SYSTÈMES DE COMMANDE PAR SIMPLE BOUCLAGE (SCSB).

I.4.1. SCSB utilisant un correcteur-S.

Le correcteur-S peut être intégré au sein de deux SCSB différents : un système utilisant un modèle de poursuite, et un système utilisant un modèle de ralliement.

I.4.1.1. Avec un modèle de poursuite.

Le système est représenté sur la figure 7 ; le correcteur est utilisé avec les arguments :

$$u(k) = \varphi_{RN}^S(y_r(k+1), y_p(k), y_p(k-1), u(k-1))$$

où la séquence $\{y_r(k)\}$ est calculée par le modèle de poursuite.

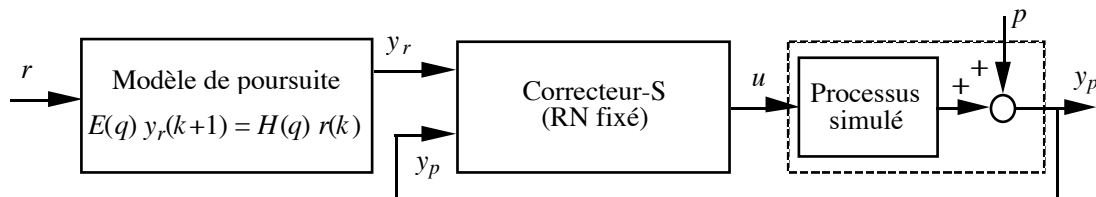


Figure 7.
SCSB utilisant un correcteur-S et un modèle de poursuite.

Les résultats du test sont rassemblés sur la figure 8a. Les défauts de modélisation accentuent les oscillations de la commande en réponse à la perturbation, par rapport au système où le correcteur est adapté au système corrigé, c'est-à-dire le modèle utilisé pour l'apprentissage (comparer à la figure 5a).

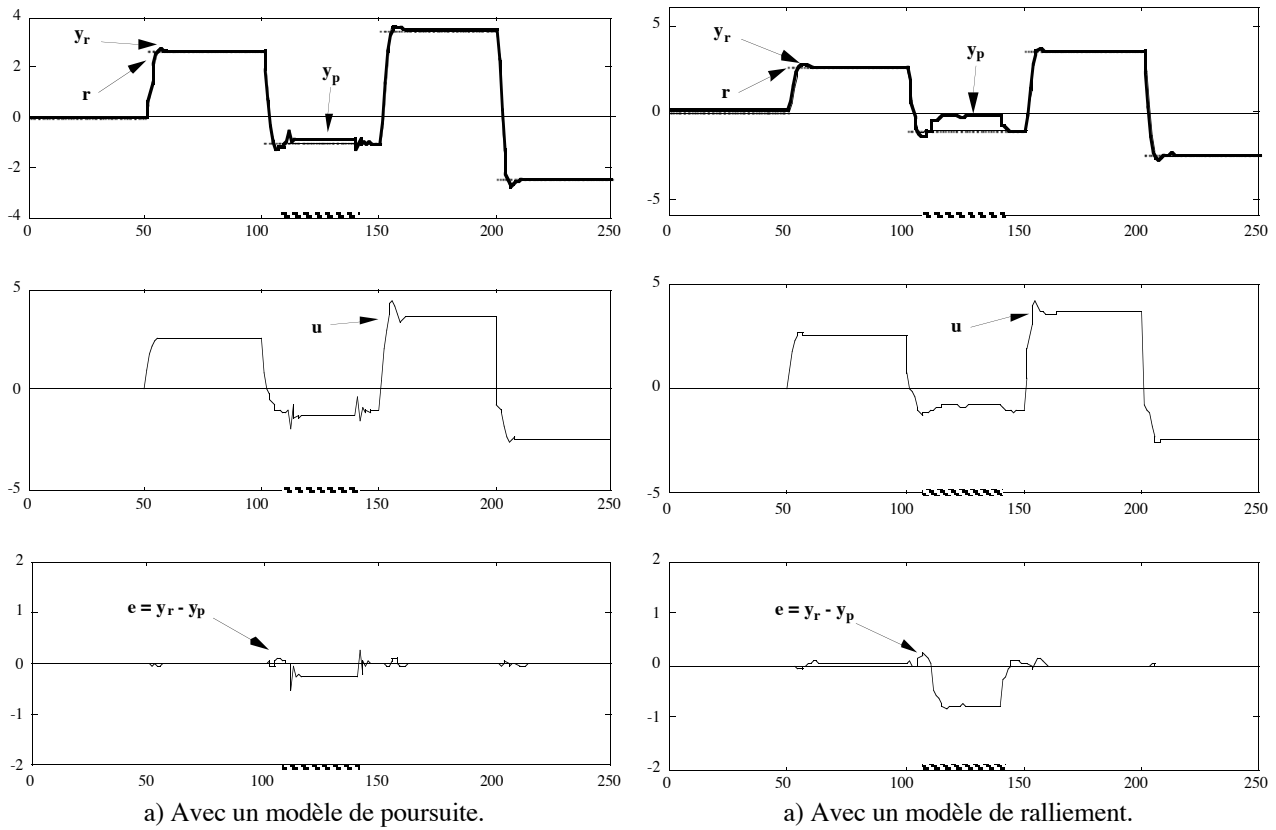


Figure 8
Performance des SCSB utilisant un correcteur-S.

I.4.1.2. Avec un modèle de ralliement.

Le système est représenté sur la figure 9 ; le correcteur-S est utilisé avec les mêmes arguments :

$$u(k) = \varphi_{RN}^S(y_r(k+1), y_p(k), y_p(k-1), u(k-1))$$

mais la séquence $\{y_r(k)\}$ est calculée par le modèle de ralliement.

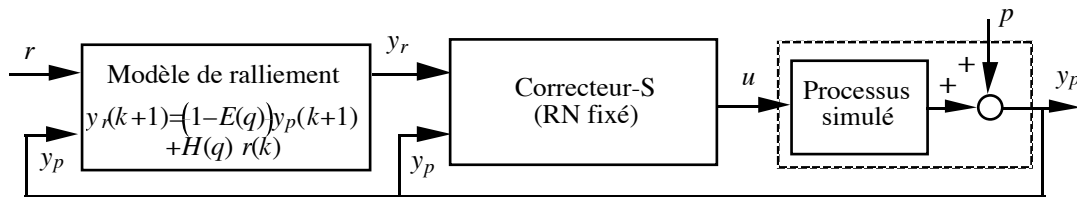


Figure 9.
SCSB utilisant un correcteur-S et un modèle de ralliement.

Les résultats sont rassemblés sur la figure 8b. Le comportement de la commande en réponse à une perturbation est satisfaisant, mais l'erreur statique reste importante (voir annexe II §II.1.1.2). De plus, il existe une erreur statique même en l'absence de perturbation (comparer à la figure 5b).

I.4.2. SCSB utilisant un correcteur-D.

Le système est représenté sur la figure 10 ; le correcteur-D est utilisé avec les arguments :

$$u(k) = \varphi_{RN}^D(H(q)r(k), y_p(k), y_p(k-1), u(k-1))$$

La séquence $\{y_r(k)\}$ est calculée par le modèle de référence $E(q) y_r(k+1) = H(q)r(k)$, et c'est sa sortie qui est utilisée pour le calcul de l'erreur, représentée en bas de la figure 11.

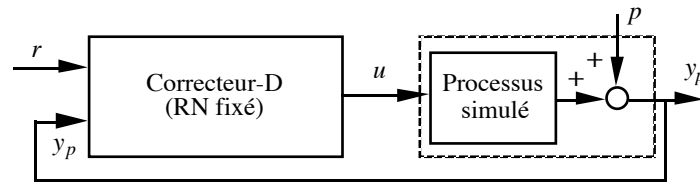


Figure 10.
SCSB utilisant un correcteur-D.

Alors que, si tout est parfait, le SCSB avec correcteur-D, et le SCSB utilisant un correcteur-S et un modèle de raliement sont équivalents, les défauts de modélisation se répercutent de façon plus importante sur le premier. En effet, la dynamique est très modifiée dans certains domaines du fonctionnement (après la perturbation notamment), et l'erreur statique en l'absence de perturbation est plus importante (dernier créneau de consigne). Ces différences sont peut-être dues à une moins bonne qualité de l'apprentissage du correcteur-D.

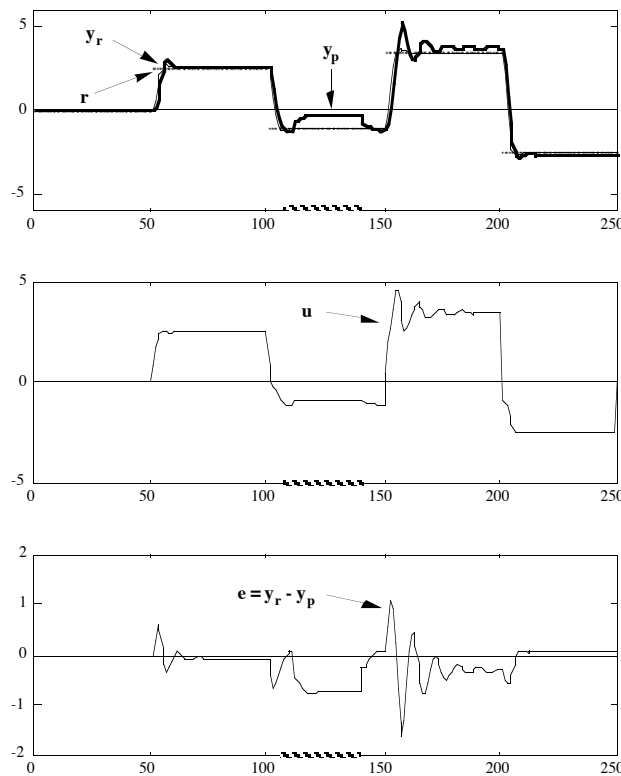


Figure 11.
Performance du SCSB utilisant un correcteur-D.

I.5. SYSTÈMES DE COMMANDE AVEC MODÈLE INTERNE (SCMI).

I.5.1. SCMI utilisant un correcteur-S.

Trois SCMI sont possibles : avec modèle de poursuite extérieur, avec modèle de poursuite intérieur, et avec modèle de raliement. Si le correcteur est parfait (ce qui est presque le cas), les deux

derniers SCMI sont équivalents, et donnent effectivement des résultats identiques. Nous ne présentons donc que le premier et le troisième de ces systèmes.

I.5.1.1. Avec un modèle de poursuite extérieur.

La figure 12 représente le SCMI ; le correcteur-S est utilisé avec les arguments :

$$u(k) = \varphi_{RN}^S(y_r^*(k+1), y(k), y(k-1), u(k-1))$$

où y est la sortie du modèle de simulation φ_{RN}^{Sim2} , utilisé comme modèle interne.

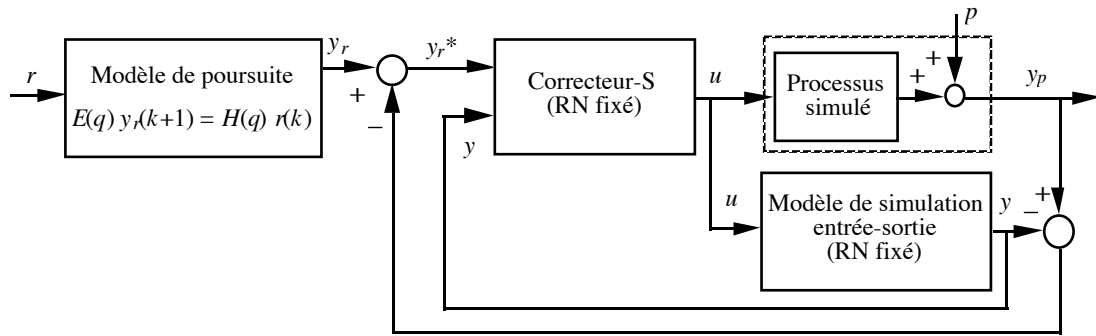


Figure 12.
SCMI utilisant un correcteur-S et un modèle de poursuite extérieur.

Les résultats sont rassemblés sur la figure 14a. Ils sont très bons puisque l'erreur statique est annulée partout, en particulier en réponse à la perturbation. Seul le comportement en régulation de la commande laisse à désirer : des oscillations se produisent.

I.5.1.2. Avec un modèle de raliement.

Le système de commande est représenté sur la figure 13.

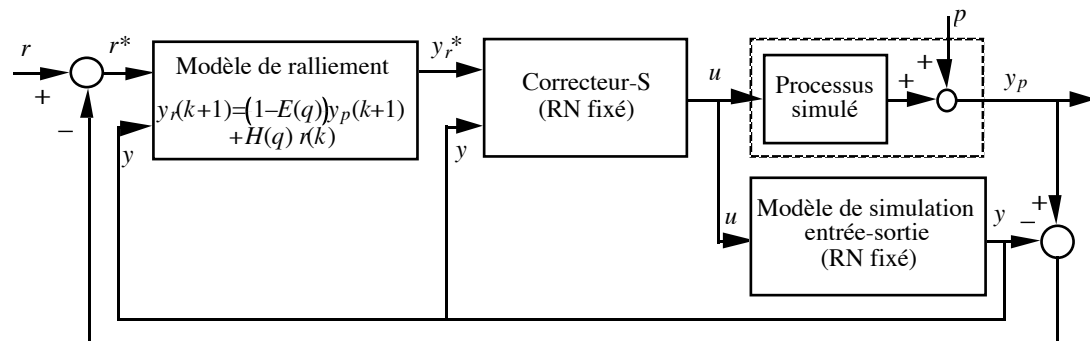


Figure 13.
SCMI utilisant un correcteur-S et un modèle de raliement.

Le correcteur est utilisé avec les mêmes arguments que dans le SCMI précédent, mais la séquence $\{y_r^*(k)\}$ est calculée par un modèle de raliement. La séquence $\{y_r(k)\}$ pour estimer l'erreur est calculée à partir de la consigne par le modèle de référence $E(q) y_r(k+1) = H(q) r(k)$ (cf. §I.4.2). Les résultats sont rassemblés sur la figure 14b. La performance est meilleure que celle du système précédent, car les oscillations de la commande en réponse à la perturbation sont filtrées par le modèle

de ralliement. Ces résultats sont analogues aux résultats établis pour un modèle linéaire dans l'annexe II §II.2.1.1 et §II.2.1.3, et généralisés aux modèles non linéaires au chapitre 5 §II.4.

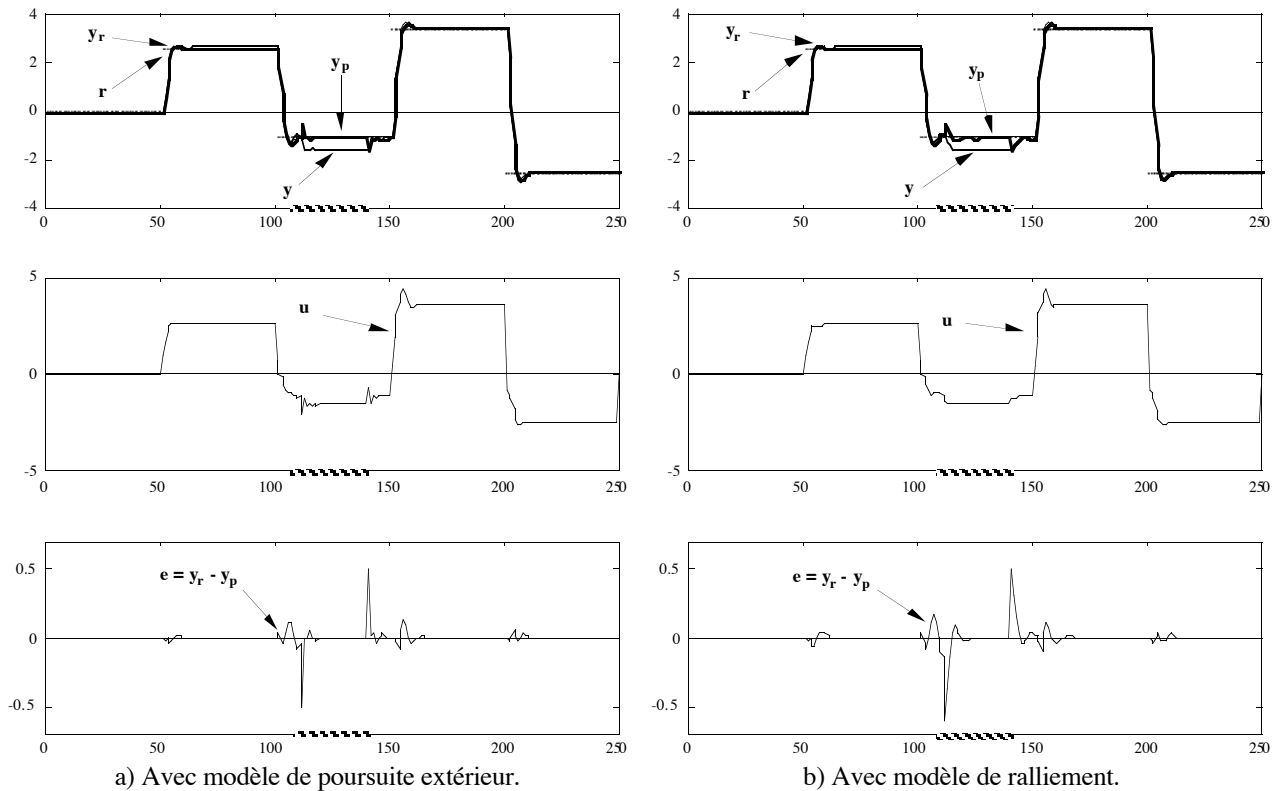


Figure 14.

Performances des SCMI avec correcteur-S.

I.5.2. SCMI utilisant un correcteur-D.

La figure 15 représente le système de commande ; le correcteur-D est utilisé avec les arguments :

$$u(k) = \varphi_{RN}^D(H(q) r^*(k), y(k), y(k-1), u(k-1))$$

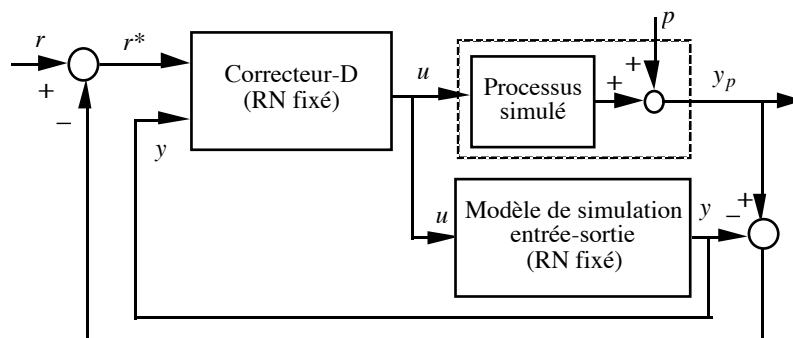


Figure 15.

SCMI utilisant un correcteur-D.

Les résultats, présentés sur la figure 16, sont tout à fait comparables à ceux de la figure 14b, conformément aux résultats de l'annexe II §II.2.1 et §II.2.2. On note cependant qu'il subsiste une petite erreur statique pendant la perturbation : ceci provient donc d'un défaut d'apprentissage du correcteur-D.

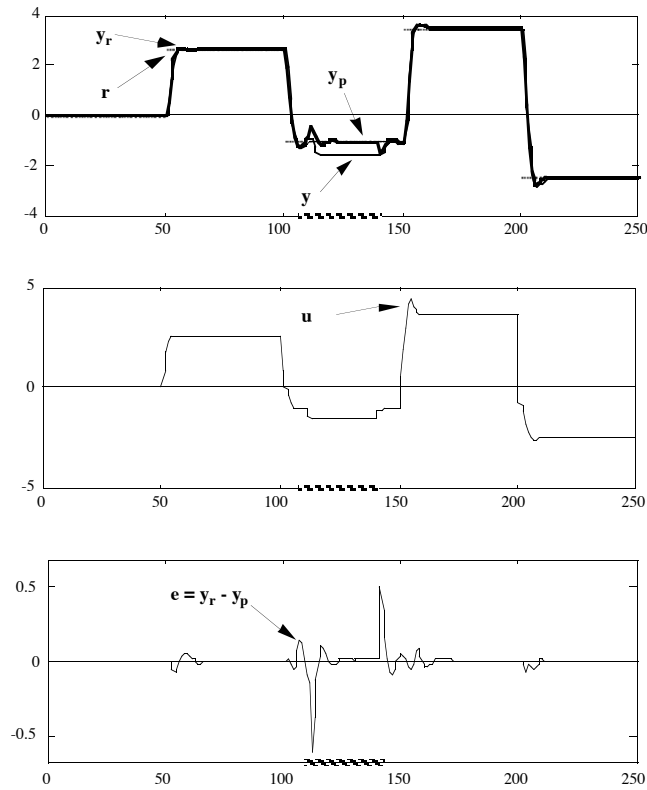


Figure 16.
Performance du SCMI utilisant un correcteur-D.

I.5.3. Suppression du modèle interne.

Au chapitre 5 §II.4.1.5, nous avons insisté sur la nécessité de conserver le modèle interne dans les SCMI, en mettant en évidence les conséquences de sa suppression, en particulier lorsque le correcteur entre en saturation.

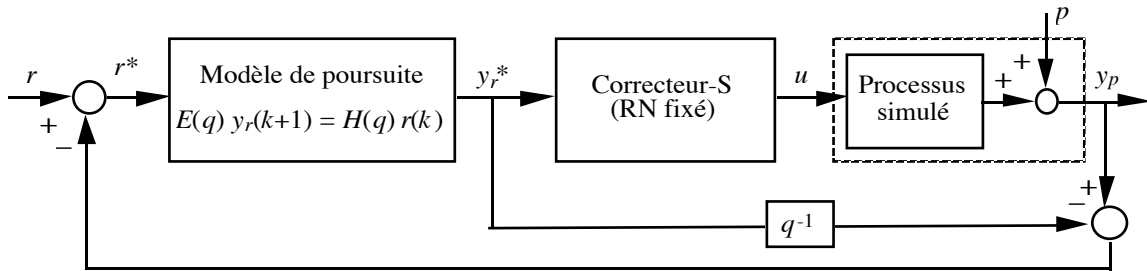


Figure 17.
“ SCMI ” utilisant un correcteur-S et un modèle de poursuite extérieur, sans MI.

Nous illustrons ceci en présentant, à l’aide d’une séquence de test analogue à la précédente, mais pour laquelle le correcteur sature (la consigne maximale est de 4,5 au lieu de 3,5), la performance du SCMI utilisant un correcteur-S et un modèle de raliement (SCMI de la figure 13), et celle du système de la figure 17 ; le correcteur-S y est utilisé avec les arguments :

$$u(k) = \varphi_{RN}^S(y_r^*(k+1), y_r^*(k), y_r^*(k-1), u(k-1))$$

Le modèle étant supprimé, le modèle de raliement doit être remplacé par un modèle de poursuite intérieur.

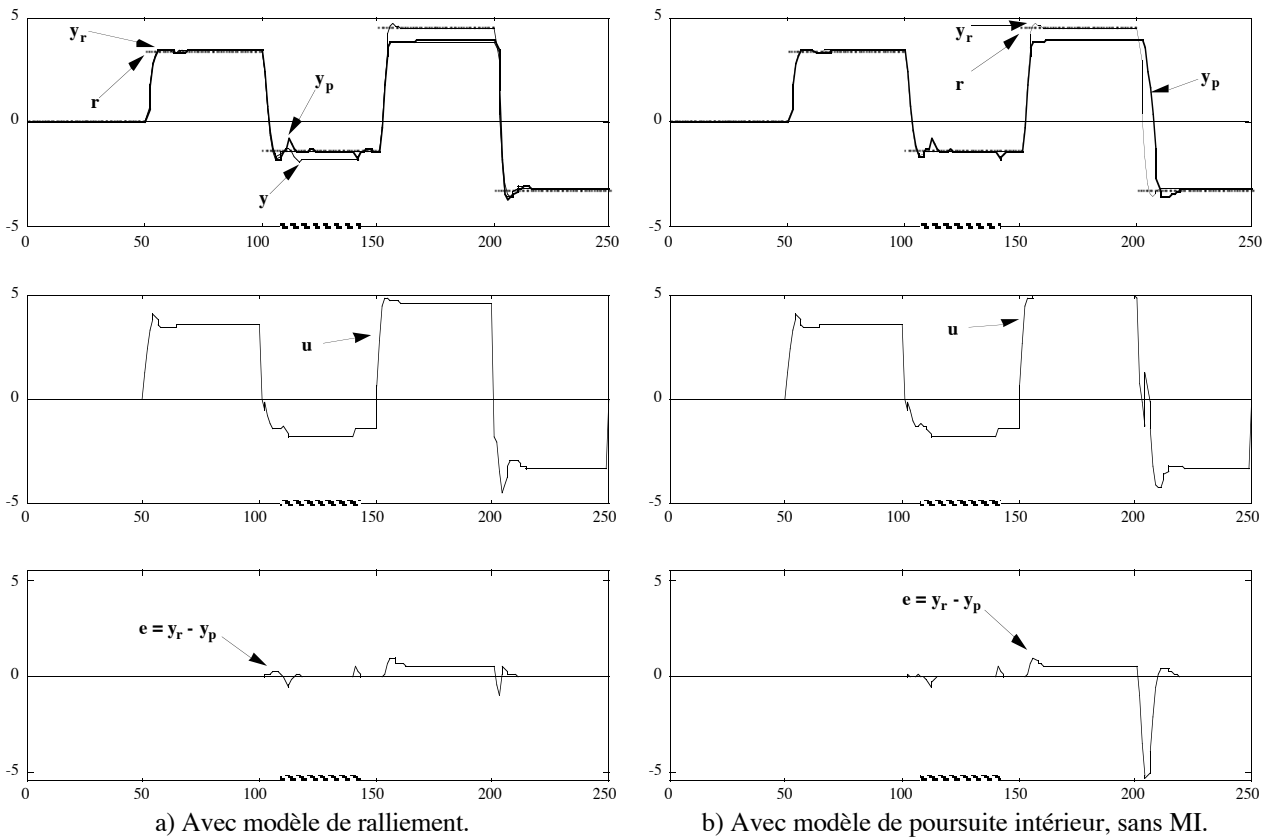


Figure 18.

Performances en saturation de SCMI utilisant un correcteur-S.

Les résultats obtenus avec les deux systèmes sont représentés sur la figure 18. Alors que le SCMI classique se comporte comme le modèle de référence dès que le correcteur ne sature plus, le système sans MI voit sa dynamique très affectée par la saturation : la boucle avec le retard unité réalise une intégration de l'erreur.

1.5.4. Cas d'un correcteur imparfait.

Dans le cas où le correcteur est parfait, les SCMI utilisant un correcteur-S avec un modèle de poursuite intérieure ou un modèle de ralliement sont équivalents. Nous avons cependant établi dans l'annexe II §II.2.1.2 que, pour un modèle linéaire, la stabilité du second système est plus robuste en cas d'une imperfection du correcteur. Afin de vérifier ceci dans un cas non linéaire, nous avons procédé à l'apprentissage d'un correcteur pour un modèle *plus lent* (la même structure de modèle, les coefficients a_1 , a_2 , b_1 et b_2 étant calculés à partir du second ordre linéaire de pulsation $\omega_n=2$, au lieu de 3). Puis nous avons utilisé ce correcteur dans les deux SCMI, avec le *même modèle de simulation* que dans les paragraphes précédents. Il s'agit donc du cas nominal (processus~modèle), avec correcteur imparfait. Le SCMI avec modèle de ralliement est celui de la figure 13, et le SCMI avec modèle de poursuite intérieure est représenté sur la figure 19 suivante.

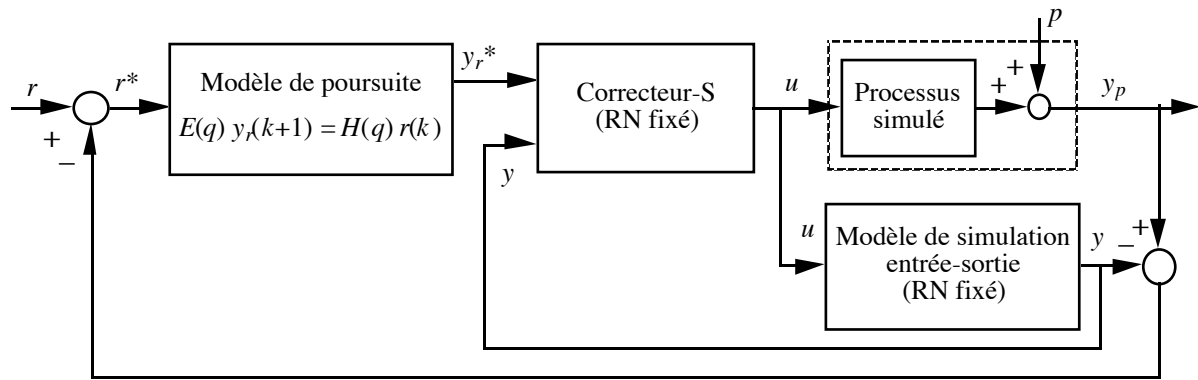


Figure 19.
SCMI utilisant un correcteur-S (*imparfait* ici) et un modèle de poursuite intérieur.

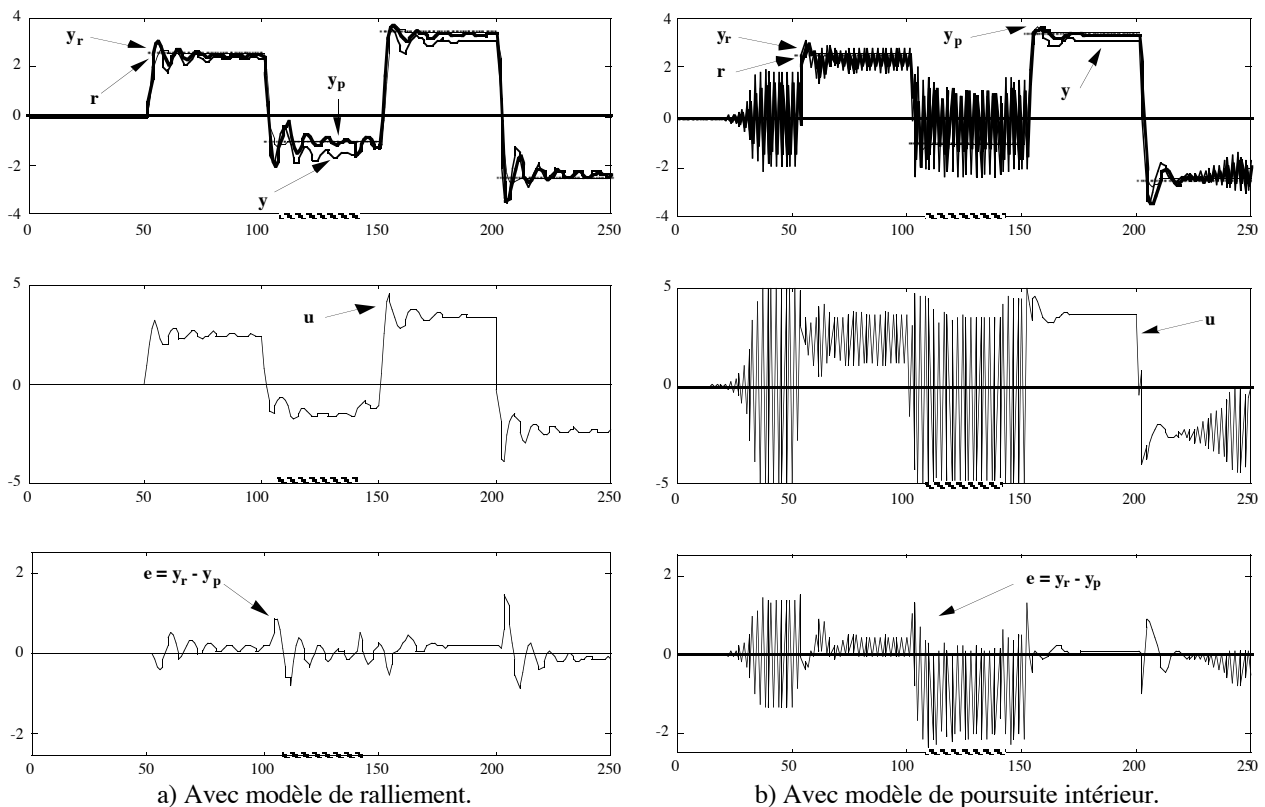


Figure 20.
SCMI utilisant un correcteur-S *imparfait*.

Les performances des deux systèmes sont rassemblées sur la figure 20. Le système avec modèle de ralliement est perturbé, mais il n'est pas déstabilisé (figure 20a). Le système avec modèle de poursuite intérieur est instable dans pratiquement tout le domaine de fonctionnement (sa sortie ne devient pas infinie car la sortie du processus est limitée par une tangente hyperbolique). Ces résultats corroborent donc la généralisation à certains systèmes non linéaires des résultats établis dans l'annexe II pour les systèmes linéaires : un SCMI neuronal doit impérativement utiliser un modèle de ralliement, en raison de l'éventuelle imperfection du correcteur due à l'apprentissage.

La performance du SCMI utilisant un correcteur-D imparfait (c'est-à-dire adapté au modèle plus lent), est tout à fait analogue à celle du SCMI utilisant un correcteur-S imparfait et un modèle de ralliement.

I.6. PROCESSUS AVEC BRUIT.

Parmi les systèmes de commande que nous avons présentés, tous ceux dont la dynamique de régulation est satisfaisante, car égale à la dynamique de poursuite, filtrent également le bruit qui peut perturber le processus (en particulier un bruit de sortie), dont nous n'avons pas tenu compte ici. Ces systèmes sont : les SCSB utilisant un correcteur-S et un modèle de ralliement, ou un correcteur-D, et les SCMI utilisant un correcteur-S et un modèle de ralliement ou un modèle de poursuite, ou un correcteur-D. Le §II a pour objet des systèmes de commande optimaux par rapport au bruit.

II. COMMANDE À VARIANCE MINIMALE.

II.1. PRÉSENTATION.

Alors que dans la première partie, nous avons privilégié la robustesse du système de commande vis-à-vis de perturbations *déterministes* non mesurées, l'objet de ce paragraphe est la synthèse de systèmes de commande optimaux vis-à-vis du bruit, c'est-à-dire des perturbations *aléatoires* non mesurées. Les systèmes de commande que nous allons présenter ont donc un intérêt pour la commande de processus pour lesquels les perturbations déterministes sont négligeables devant les perturbations aléatoires. Nous reprenons l'exemple du processus simulé par le modèle entrée-sortie du chapitre 4, processus dont la partie déterministe est :

$$y_p(k) = h(y_p(k-1), y_p(k-2), u(k-1)) = 50 \tanh \left[2 \cdot 10^{-3} \left(\frac{(24 + y_p(k-1))}{3} y_p(k-1) - 8 \frac{u(k-1)^2}{1 + u(k-1)^2} y_p(k-2) \right) \right] + 0,5 u(k-1)$$

Les systèmes de commande seront conçus pour :

* le processus avec perturbation additive d'état (NARX) :

$$y_p(k) = h(y_p(k-1), y_p(k-2), u(k-1)) + w(k)$$

* le processus avec perturbation additive de sortie (NBSX) :

$$\begin{cases} x_p(k) = h(x_p(k-1), x_p(k-2), u(k-1)) \\ y_p(k) = x_p(k) + w(k) \end{cases}$$

où w un bruit blanc à valeur moyenne nulle et de distribution uniforme d'amplitude 0,2 (donc de variance $3,33 \cdot 10^{-3}$).

II.2. SYSTÈMES DE COMMANDE À VARIANCE MINIMALE.

Pour un processus sans bruit, le correcteur qui minimise la variance de l'erreur de commande est justement le correcteur-S : il annule même l'erreur de commande. Dans le cas d'un processus avec bruit, nous allons montrer que, dans les cas NARX et NBSX, le système de commande minimisant la variance de l'erreur s'obtient facilement à partir du correcteur-S associé au modèle sans bruit. Étudions préalablement le correcteur à variance minimale d'un modèle avec bruit linéaire.

II.2.1. Cas linéaire ARMAX.

Pour cette présentation rapide, nous confondons le modèle du processus et le processus lui-même.

Correcteur à variance minimale (processus avec bruit).

Soit un processus ARMAX :

$$A(q) y_p(k) = B(q) u(k) + C(q) w(k) = q^{-d} B'(q) u(k) + C(q) w(k)$$

où $w(k)$ est un bruit blanc, et C un polynôme monique de degré p . On montre [GOO84] que la commande à variance minimale est :

$$u(k) = y_r(k+d) - G_b(q) y_p(k) + (1 - F_b(q) B'(q)) u(k) + (C(q) - 1) y(k+d) \quad (\text{I})$$

où $y(k+d)$ est la prédiction optimale à l'instant k de $y_p(k+d)$ (dont l'expression est établie au chapitre 2 §II.2), et où F_b , monique, et G_b sont les uniques polynômes de degrés respectifs $d-1$ et $n-1$ satisfaisant : $C(q) = F_b(q) A(q) + q^{-d} G_b(q)$. Le système obéit alors à :

$$e(k+d) = y_p(k+d) - y_r(k+d) = F_b(q) w(k+d)$$

soit, si la commande est appliquée à partir de l'instant 0, $e(k) = F_b(q) w(k)$ quel que soit $k \geq d$.

Correcteur-S (processus sans bruit).

Rappelons l'expression du correcteur pour le processus déterministe (annexe II §I.1.1) :

$$A(q) y_p(k) = B(q) u(k) = q^{-d} B'(q) u(k)$$

L'expression du correcteur-S est :

$$u(k) = y_r(k+d) - G(q) y_p(k) + (1 - F(q) B'(q)) u(k) \quad (\text{II})$$

où F , monique, et G sont les uniques polynômes de degrés respectifs $d-1$ et $n-1$ satisfaisant l'égalité polynômiale : $1 = F(q) A(q) + q^{-d} G(q)$. Dans le cas où $d=1$, $F(q) = 1$; $G(q) = q(1 - A(q))$, et (II) devient :

$$u(k) = y_r(k+1) + (A(q) - 1) y_p(k+1) + (1 - B'(q)) u(k)$$

On constate que :

- dans le cas particulier ARX, la commande calculée par le correcteur-S est le correcteur à variance minimale. En effet, $C(q) = 1$ et les lois I et II sont donc identiques. Ceci est vrai quel que soit le retard d .
- dans le cas particulier BSX, $C(q) = A(q)$ et donc $F_b(q) = 1$; $G_b(q) = 0$. La commande I s'écrit :

$$u(k) = y_r(k+d) + (1 - B'(q)) u(k) + (A(q) - 1) y(k+d)$$
 qui, si $d=1$, est aussi l'expression du correcteur-S, mais appliquée à la sortie $y(k+1)$ du prédicteur optimal et non à celle du processus.

II.2.2. Cas non linéaires NARX (d quelconque) et NBSX (d=1).

De même, dans le cas non linéaire :

- le correcteur à variance minimale pour un processus NARX de retard quelconque est le correcteur-S du processus sans bruit.
- le correcteur à variance minimale pour un processus NBSX avec $d=1$ est constitué du correcteur-S du processus sans bruit, et du prédicteur optimal NBSX.

Le cas NARMAX dans le cas particulier d'un bruit MA est également facile à traiter, et exposé rapidement dans [GOO84].

II.3. MISE EN ŒUVRE.

II.3.1. Apprentissage du correcteur-S.

Les systèmes de commande à variance minimale des processus NARX et NBSX avec $d=1$ ne nécessitent donc que l'apprentissage du correcteur-S pour processus sans bruit.

Modèle de simulation.

Le modèle de simulation est le modèle φ_{RN}^{Sim1} obtenu à partir du prédicteur optimal NBSX (identifié au chapitre 4 §I.3.2.1) :

$$y(k) = \varphi_{RN}^{Sim1}(y(k-1), y(k-2), u(k-1))$$

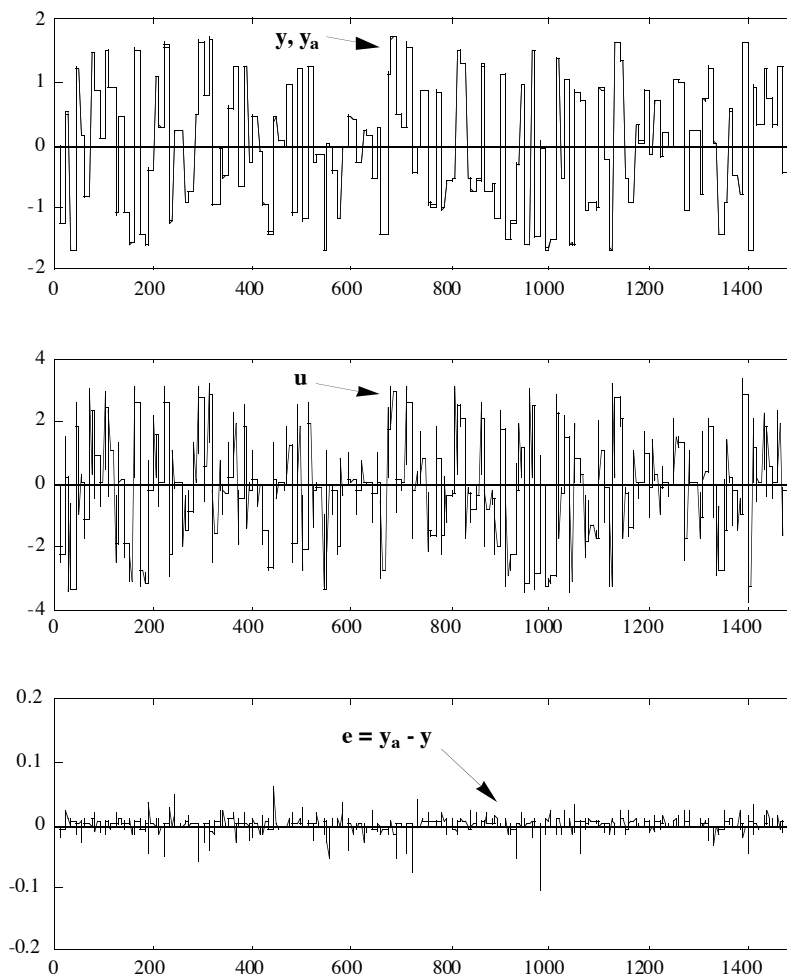


Figure 21.
Performances sur la séquence d'apprentissage.

Correcteur-S.

Nous prenons un réseau de neurones de même complexité que celui du modèle de simulation, possédant 5 neurones cachés complètement connectés, et un neurone de sortie à sigmoïde d'amplitude

5, qui est la valeur maximale de la commande utilisée pour l'identification, valeur qui ne doit donc pas être dépassée pendant l'apprentissage du correcteur. Le correcteur-S est un réseau non bouclé :

$$u(k) = \psi_{RN}(r(k), y(k), y(k-1); C)$$

Modèle de référence d'apprentissage.

C'est un retard unité. Sa sortie y_a obéit à : $y_a(k+1) = r(k)$

Algorithme d'apprentissage.

L'algorithme est semi-dirigé.

Séquences d'apprentissage.

La séquence de consigne est constituée de créneaux d'amplitudes aléatoires entre $\pm 1,75$ (sorties maximales du processus pour une commande entre ± 5), et de durée 10 pas d'échantillonnage. La séquence totale comporte 1500 pas d'échantillonnage. Les résultats de l'apprentissage sont représentés sur la figure 21.

Séquences d'estimation de la performance des systèmes de commande.

La séquence de consigne est constituée de créneaux d'amplitudes aléatoires comprises entre $\pm 1,75$, et de durée 20 pas d'échantillonnage. La séquence totale comporte 1000 pas d'échantillonnage (cf. figures 23 et 25).

II.3.2. Système de commande du processus NARX.

Le processus NARX est simulé par l'équation :

$$y_p(k) = h(y_p(k-1), y_p(k-2), u(k-1)) + w(k)$$

où w est un bruit blanc à valeur moyenne nulle, de distribution uniforme d'amplitude 0,2. Le correcteur-S obtenu en fin d'apprentissage, noté ψ_{RN}^S , est mis en cascade avec le processus :

$$u(k) = \varphi_{RN}^S(y_r(k+1), y_p(k), y_p(k-1))$$

Le système de commande à variance minimale pour le processus NARX est représenté sur la figure 22. Le comportement désiré en poursuite est donné par un modèle de référence qui est la discrétisation du filtre passe-bas du second ordre de pulsation $\omega_n=0,5$, de coefficient d'amortissement $\xi=0,7$, et de gain unité. Ce filtre passe-bas est toujours plus rapide que le processus (dans tout l'espace d'état). Il est noté :

$$E(q) y_r(k+1) = H(q) r(k)$$

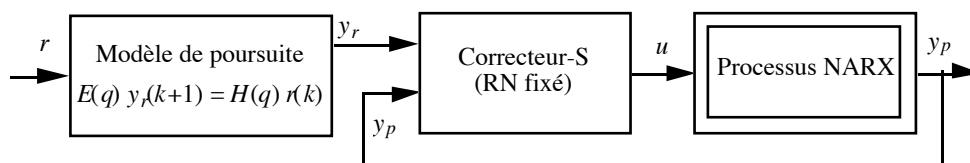


Figure 22.
Système de commande à variance minimale pour le processus NARX.

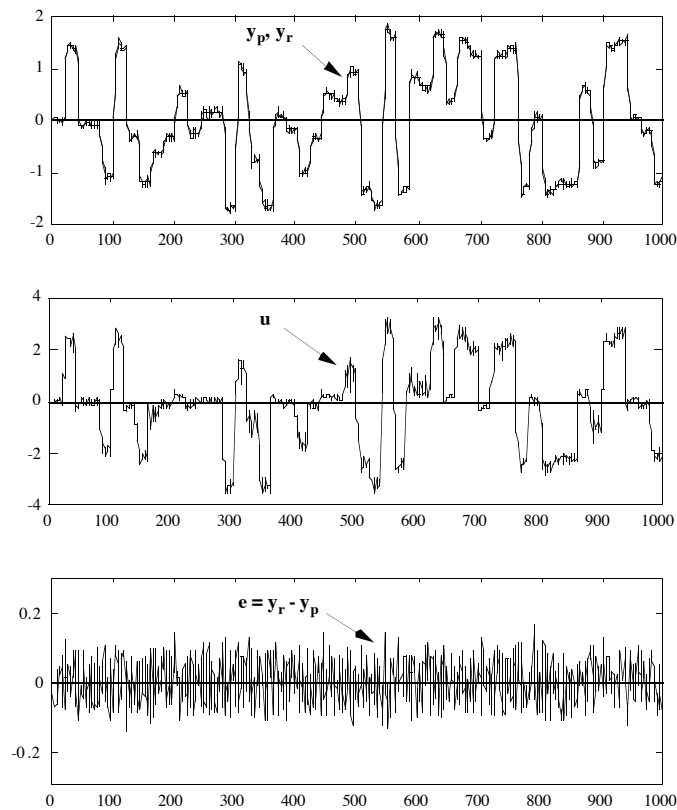


Figure 23.

Performance du système de commande à variance minimale du processus NARX.

Les résultats obtenus sont représentés sur la figure 23. L'erreur de commande a bien les mêmes caractéristiques que la perturbation aléatoire w , c'est-à-dire qu'elle n'est pas corrélée et que sa distribution est semblable à celle de la perturbation (uniforme d'amplitude 0,2).

II.3.3. Systèmes de commande du processus NBSX.

Le processus NBSX est simulé par l'équation :

$$\begin{cases} x_p(k) = h(x_p(k-1), x_p(k-2), u(k-1)) \\ y_p(k) = x_p(k) + w(k) \end{cases}$$

où w est un bruit blanc à valeur moyenne nulle, de distribution uniforme d'amplitude 0,2. Le correcteur-S obtenu en fin d'apprentissage est mis en cascade avec le prédicteur optimal du processus NBSX, qui est le prédicteur bouclé :

$$y(k+1) = \varphi_{RN}^{Sim1}(y(k), y(k-1), u(k))$$

Le correcteur est donc utilisé avec les arguments :

$$u(k) = \varphi_{RN}^S(y_r(k+1), y(k), y(k-1))$$

Le système de commande est représenté sur la figure 24.

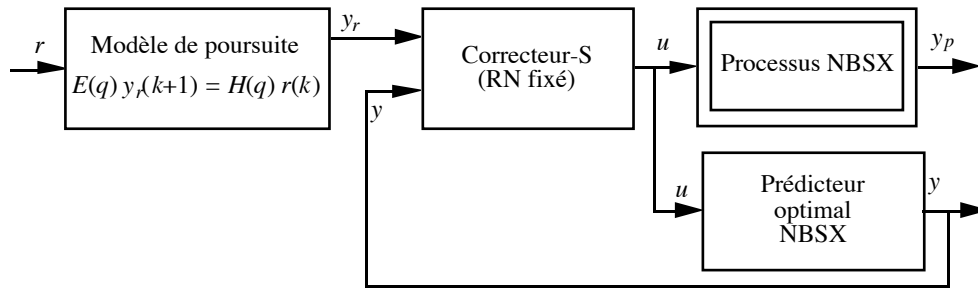


Figure 24.
Système de commande à variance minimale pour le processus NBSX.

Les résultats obtenus sont représentés sur la figure 25a. Ici encore, l'erreur de commande a bien les mêmes caractéristiques que la perturbation aléatoire (distribution, corrélation). On constate que la commande n'est pas bruitée, puisque calculée à partir des sorties du *prédicteur*, qui est déterministe.

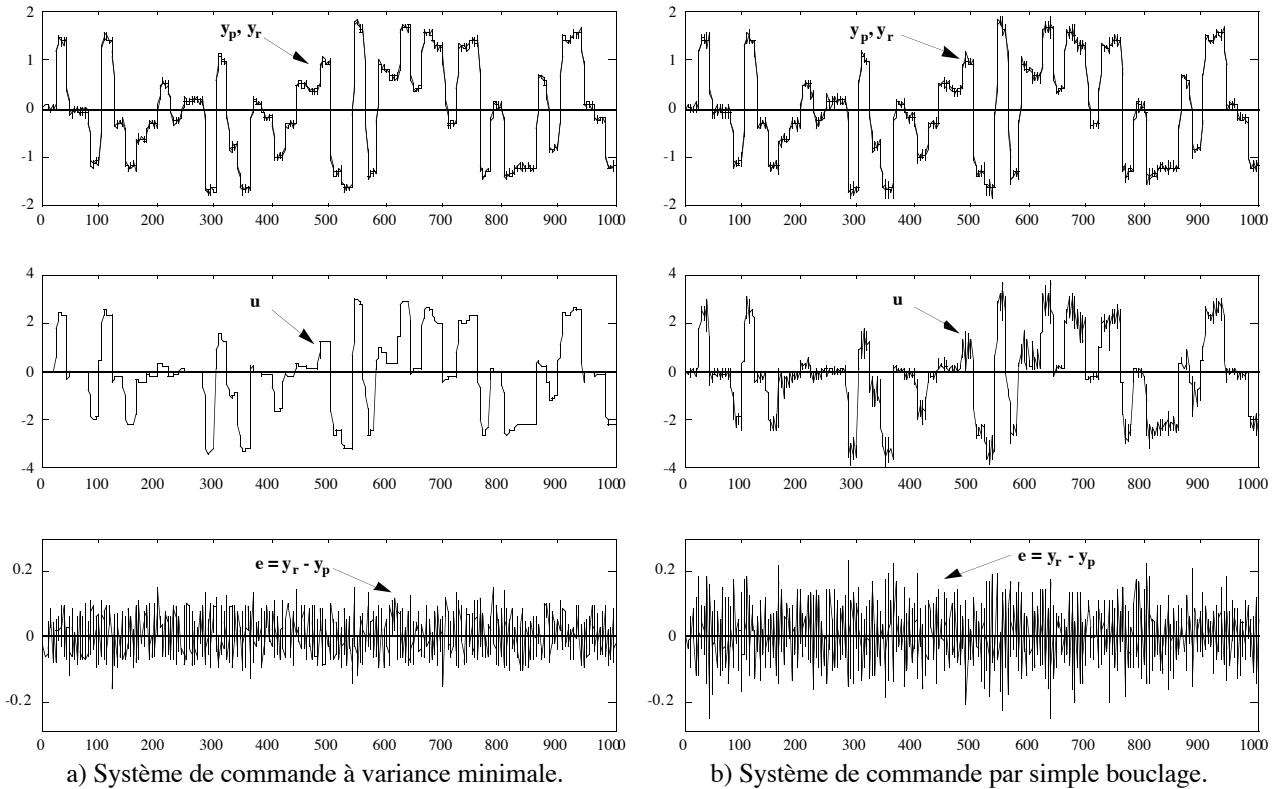


Figure 25.
Performances des systèmes de commande du processus NBSX .

À titre de comparaison, nous avons également utilisé le correcteur-S en simple bouclage avec le processus (même système de commande que celui de la figure 22, adéquat à la commande du processus NARX). Les résultats sont représentés sur la figure 25b. La variance de l'erreur de commande est plus importante que celle de la perturbation aléatoire qui affecte le processus, et cette erreur n'a plus les caractéristiques d'un bruit blanc.

CONCLUSION.

La première partie de ce chapitre nous a permis d'illustrer, sur l'exemple d'un processus non linéaire, les propriétés des systèmes de commande par simple bouclage et avec modèle interne démontrées dans le cas linéaire en annexe II. Elle fait notamment apparaître les propriétés de robustesse des systèmes préconisés, c'est-à-dire les systèmes avec modèle interne utilisant un correcteur-S avec un modèle de ralliement, ou bien un correcteur-D, vis-à-vis :

- de perturbations déterministes de sortie en créneaux ;
- de défauts de modélisation (le modèle de simulation utilisé est identifié dans de mauvaises conditions au chapitre 4 §II.3.2.2) ;
- de défauts du correcteur dus à l'apprentissage, problème particulier à l'utilisation de réseaux de neurones.

De plus, nous mettons en évidence le manque de robustesse de systèmes de commande souvent rencontrés dans la littérature.

Dans la seconde partie, nous jetons les bases du principe de la commande à variance minimale par réseaux de neurones pour des processus NARX (de retard quelconque) et NBSX (de retard unité). Un axe de nos recherches futures porte sur la généralisation de ce type de commande aux processus NARMAX.

