

Une méthodologie pour l'amélioration de la convergence des algorithmes à erreur de sortie

Elie TOHME^{1,2}, Régis OUVRARD¹, Jean-Claude TRIGEASSOU¹, Antoine ABCHE², Thierry POINOT¹

¹Laboratoire d'Automatique et d'Informatique Industrielle
Ecole Supérieure d'Ingénieurs de Poitiers, 40, avenue du Recteur Pineau 86022 Poitiers Cedex, France

²Faculty of Engineering, University of Balamand, P.O. Box 100 Tripoli, Lebanon

elie.tohme@etu.univ-poitiers.fr, regis.ouvrard@univ-poitiers.fr,
jean-claude.trigeassou@univ-poitiers.fr, antoine.abche@alamand.edu.lb,
thierry.poinot@univ-poitiers.fr

Résumé— Dans cette communication, un nouvel algorithme d'optimisation hors-ligne est proposé pour tendre vers une convergence globale. Cet algorithme, appelé algorithme à Pseudo-Erreur de Sortie (PES), est basé sur l'introduction d'un filtre stationnaire dans les fonctions de sensibilité paramétrique de l'algorithme de Gauss-Newton. La convergence globale est assurée pour un système du premier ordre quelque soit le filtre stationnaire du même ordre. Par conséquent, une décomposition en éléments parallèles du premier ordre d'un système d'ordre n est introduite. L'identification de chaque élément du premier ordre par l'algorithme PES est globalement convergente. La décomposition fait intervenir un filtrage, et nécessite une connaissance a priori sur les paramètres, qui peuvent biaiser l'estimation. Cependant, les paramètres obtenus sont voisins des paramètres réels et sont alors utilisés pour initialiser l'algorithme à erreur de sortie de Gauss-Newton. Les résultats de simulation montrent l'efficacité de cette approche. En particulier, alors que l'algorithme de Gauss-Newton diverge à partir d'une certaine initialisation, l'algorithme PES, appliqué sur les éléments issus de la décomposition, fournit des paramètres utilisés pour l'initialisation de l'algorithme de Gauss-Newton qui permettent la convergence vers l'optimum global.

Mots-clés— Algorithme à erreur de sortie, Algorithme à pseudo-erreur de sortie, Convergence globale, Décomposition en élément du premier ordre.

I. INTRODUCTION

Les algorithmes d'identification par Erreur de Sortie (ES) présentent l'intérêt essentiel de fournir, sous certaines hypothèses, une estimation non biaisée asymptotiquement [1][2]. Malheureusement, cette propriété est obtenue au prix d'une minimisation par programmation non linéaire du critère quadratique, ce qui se traduit généralement par la non unicité de l'optimum [3][4]. Ce problème fondamental peut être partiellement résolu en réalisant des optimisations à partir de différents points initiaux pour "cartographier" le critère quadratique et ainsi choisir l'optimum parmi les résultats obtenus. Néanmoins, cette démarche est très coûteuse en temps de calcul.

Une autre approche, très répandue, revient à initialiser l'algorithme à erreur de sortie par une optimisation linéaire (algorithme de type moindres carrés) à l'aide d'une technique à erreur d'équation. Cette première estimation est asymptotiquement biaisée, mais généralement voisine de l'optimum

global recherché. Cette méthodologie peut être affinée et perfectionnée grâce à une technique de variable instrumentale. Citons par exemple la méthode utilisant les moments partiels réinitialisés [5][6] qui fournit une initialisation particulièrement efficace en terme de convergence.

Le problème de la convergence en erreur de sortie peut être résolu par une approche différente dans son principe. Landau [7] utilise pour cela un algorithme récursif déterminé à l'aide de la théorie de l'hyperstabilité. L'analyse de cet algorithme permet de démontrer sa convergence vers un optimum unique. L'apparente contradiction avec l'approche générale vient du fait que cet algorithme récursif ne minimise pas de critère quadratique. Lors de précédents travaux [8][9], un algorithme hors ligne généralisant l'approche précédente a été proposé. Cet algorithme, appelé algorithme à Pseudo-Erreur de Sortie (PES), est présenté dans [10].

La particularité de l'algorithme PES, comme celui de Landau, est d'utiliser un filtre stationnaire dans la procédure itérative d'optimisation. La convergence globale de cet algorithme est assurée si le filtre stationnaire respecte une certaine hypothèse de passivité. Cette hypothèse repose sur les pôles réels du système qui sont recherchés et donc inconnus. Cependant, cette hypothèse est toujours vraie pour un système et un filtre du premier ordre. L'idée clé de l'algorithme, présenté dans cette communication, est de décomposer le système d'ordre n à identifier en éléments parallèles du premier ordre et d'identifier chaque élément par l'algorithme PES. L'hypothèse de convergence globale est vérifiée pour l'estimation de chaque élément. Malheureusement, la décomposition fait intervenir un filtrage, et nécessite une connaissance a priori sur les paramètres, qui peuvent biaiser l'estimation. Cependant, les paramètres obtenus sont voisins des paramètres réels et sont alors utilisés pour initialiser l'algorithme à erreur de sortie de Gauss-Newton. Remarquons que cette approche ne s'applique qu'à des systèmes à pôles réels.

Une présentation de l'algorithme PES est donnée dans la section II. La décomposition en éléments du premier ordre est présentée dans la section III. Enfin, la section IV illustre cette approche avec deux applications en simulation. La section V conclut ce papier.

Pour faciliter la présentation de la décomposition en éléments du premier ordre, les développements ci-dessous sont appliqués à un système à temps continu. Le passage à un système à temps discret est trivial.

II. L'ALGORITHME À PSEUDO-ERREUR DE SORTIE

A. Le système réel et le modèle

Considérons le système stable d'ordre n composé de pôles réels et défini par sa fonction de transfert

$$H(s) = \frac{B(s)}{A(s)} \quad (1)$$

où

$$\begin{aligned} A(s) &= a_0 + a_1s + \dots + a_{n-1}s^{n-1} + s^n \\ B(s) &= b_0 + b_1s + \dots + b_ms^m, \quad m < n \end{aligned} \quad (2)$$

$y(t)$ et $u(t)$ sont respectivement la sortie et l'entrée du système.

Soit $\hat{H}(s, \hat{\theta})$ le modèle de même structure que le système

$$\hat{H}(s, \hat{\theta}) = \frac{\hat{b}_0 + \hat{b}_1s + \dots + \hat{b}_ms^m}{\hat{a}_0 + \hat{a}_1s + \dots + \hat{a}_{n-1}s^{n-1} + s^n} \quad (3)$$

avec

$$\hat{\theta} = [\hat{a}_0, \dots, \hat{a}_{n-1}, \hat{b}_0, \dots, \hat{b}_m]^T \quad (4)$$

$y_M(t, \hat{\theta})$ est la sortie du modèle en réponse à l'excitation $u(t)$.

B. L'algorithme à Erreur de Sortie

On dispose de N mesures de l'entrée et de la sortie.

L'algorithme ES considéré est l'algorithme de Gauss-Newton appliqué au critère quadratique suivant

$$\begin{aligned} J(\hat{\theta}) &= \frac{1}{N} \sum_{t=1}^N (y(t) - y_M(t, \hat{\theta}))^2 \\ &= \frac{1}{N} \sum_{t=1}^N \varepsilon_M(t)^2 \end{aligned} \quad (5)$$

Définissons les fonctions de sensibilité paramétrique

$$\begin{aligned} \sigma(t, \hat{\theta}) &= [\sigma_{\hat{a}_0}(t), \dots, \sigma_{\hat{a}_{n-1}}(t), \sigma_{\hat{b}_0}(t), \dots, \sigma_{\hat{b}_m}(t)]^T \\ \sigma_{\hat{a}_j}(t) &= \frac{\partial}{\partial \hat{a}_j} y_M(t, \hat{\theta}), \quad \sigma_{\hat{b}_j}(t) = \frac{\partial}{\partial \hat{b}_j} y_M(t, \hat{\theta}) \end{aligned} \quad (6)$$

L'algorithme de Gauss-Newton standard est défini par

$$\hat{\theta}_{i+1} = \hat{\theta}_i - \left\{ \left[J''_{\hat{\theta}} \right]^{-1} J'_{\hat{\theta}} \right\}_{\hat{\theta}=\hat{\theta}_i} \quad (7)$$

où le gradient $J'_{\hat{\theta}}$ et le Hessian $J''_{\hat{\theta}}$ sont définis comme suit

$$J'_{\hat{\theta}} = \frac{\partial J}{\partial \hat{\theta}} = -\frac{2}{N} \sum_{t=1}^N \sigma(t, \hat{\theta}) \varepsilon_M(t) \quad (8)$$

$$J''_{\hat{\theta}} = \frac{\partial^2 J}{\partial \hat{\theta} \partial \hat{\theta}^T} \approx \frac{2}{N} \sum_{t=1}^N \sigma(t, \hat{\theta}) \sigma^T(t, \hat{\theta}) \quad (9)$$

Notons que

$$\begin{aligned} \frac{\partial}{\partial \hat{a}_j} Y_M(s, \hat{\theta}) &= \frac{-s^j (\hat{b}_0 + \hat{b}_1s + \dots + \hat{b}_ms^m)}{(\hat{a}_0 + \hat{a}_1s + \dots + \hat{a}_{n-1}s^{n-1} + s^n)^2} U(s) \\ \frac{\partial}{\partial \hat{b}_j} Y_M(s, \hat{\theta}) &= \frac{s^j}{\hat{a}_0 + \hat{a}_1s + \dots + \hat{a}_{n-1}s^{n-1} + s^n} U(s) \end{aligned} \quad (10)$$

où $Y_M(s, \hat{\theta})$ et $U(s)$ sont respectivement les transformées de Laplace de $y_M(t, \hat{\theta})$ et $u(t)$.

La transformée de Laplace $\sigma(s, \hat{\theta})$ du vecteur des fonctions de sensibilité $\sigma(t, \hat{\theta})$ peut être mise sous la forme

$$\begin{aligned} \sigma(s, \hat{\theta}) &= \frac{1}{A(s)} \Phi_M(s, \hat{\theta}) \\ \hat{A}(s) &= \hat{a}_0 + \hat{a}_1s + \dots + \hat{a}_{n-1}s^{n-1} + s^n \\ \Phi_M(s, \hat{\theta}) &= \begin{bmatrix} -Y_M(s, \hat{\theta}), \dots, -s^{n-1}Y_M(s, \hat{\theta}), \\ U(s), \dots, s^m U(s) \end{bmatrix}^T \end{aligned} \quad (11)$$

C. L'algorithme à Pseudo-Erreur de Sortie

L'originalité des algorithmes à erreur de sortie proposés par Landau [7] réside dans l'utilisation d'un filtre stationnaire sur les résidus. Ces algorithmes présentent la propriété d'être globalement asymptotiquement stables sous certaines hypothèses de passivité. L'idée [8][9] est de modifier les fonctions de sensibilité définies par (11) en remplaçant le filtre $1/\hat{A}(s)$ par un filtre stationnaire $1/D(s)$. Ainsi la propriété de convergence globale des algorithmes récursifs est étendue aux algorithmes hors-ligne.

Par conséquent, définissons ces pseudo-fonctions de sensibilité

$$\phi(s, \hat{\theta}) = \frac{1}{D(s)} \Phi_M(s, \hat{\theta}) \quad (12)$$

où $D(s)$ est un filtre judicieusement choisi, et $\phi(s, \hat{\theta})$ est la transformée de Laplace du signal temporel $\phi(t, \hat{\theta})$.

Il en découle un pseudo-gradient et un pseudo-Hessian définis respectivement par

$$\tilde{J}'_{\hat{\theta}} = -\frac{2}{N} \sum_{t=1}^N \phi(t, \hat{\theta}) \varepsilon_M(t) \quad (13)$$

$$\tilde{J}''_{\hat{\theta}} = \frac{2}{N} \sum_{t=1}^N \phi(t, \hat{\theta}) \phi^T(t, \hat{\theta}) \quad (14)$$

En substituant $\sigma(t, \hat{\theta})$ par $\phi(t, \hat{\theta})$ dans l'algorithme ES de Gauss-Newton (7), on obtient l'algorithme à Pseudo-Erreur de Sortie donné par

$$\hat{\theta}_{i+1} = \hat{\theta}_i + \left\{ \left[\sum_{t=1}^N \phi(t, \hat{\theta}) \phi^T(t, \hat{\theta}) \right]^{-1} \sum_{t=1}^N \phi(t, \hat{\theta}) \varepsilon_M(t) \right\}_{\hat{\theta}=\hat{\theta}_i} \quad (15)$$

Remarque 1: A première vue, (7) et (15) apparaissent semblables. Cependant, il y a une différence fondamentale dans le mécanisme de convergence des deux algorithmes. Dans l'algorithme ES, l'estimation est réalisée par minimisation du critère $J(\hat{\theta})$. Dans l'algorithme PES, l'introduction du filtre

stationnaire fait que l'algorithme converge vers un optimum unique ; en fait, il n'y plus de minimisation de critère. C'est une façon radicale de se débarrasser des optimums secondaires. C'est la différence essentielle entre ces deux algorithmes. Par contre, on devine facilement que le choix du filtre stationnaire $1/D(s)$ est une étape cruciale de l'algorithme PES.

Remarque 2: L'algorithme PES est basé sur l'algorithme de Gauss-Newton. Une extension à un algorithme relaxé, tel que l'algorithme de Marquardt, est envisageable. Cette extension est écartée dans cet article, car l'objectif est de montrer l'efficacité de l'algorithme PES sur l'algorithme ES sans qu'un terme de relaxation vienne masquer les problèmes de convergence. Bien sûr, en pratique, un algorithme PES de type Marquardt peut être utilisé.

D. Choix du filtre stationnaire $1/D(s)$

Le filtre stationnaire idéal pour assurer une convergence globale est un filtre ayant les paramètres exacts du système réel $D(s) = A(s)$. Seulement, ces paramètres sont inconnus puisqu'on cherche à les identifier. Généralement, il est possible d'obtenir, à l'aide d'une réponse indicelle ou d'une étude harmonique, une connaissance a priori sur la bande passante du système réel. On choisit alors un filtre passe-bas avec pour bande passante, celle connue a priori, et pour ordre, l'ordre du système.

On peut imaginer ensuite différentes techniques pour revenir sur ce choix durant la procédure itérative, et ainsi, améliorer l'algorithme.

D'autres filtres peuvent être utilisés, comme par exemple le filtre à réponse impulsionnelle finie tiré des moments partiels réinitialisés [5][11].

E. Convergence de l'algorithme PES

La convergence de l'algorithme PES a été étudiée en utilisant la technique de l'ODE proposée par Ljung [12][13]. Les détails de cette étude sont donnés dans [10]. Pour assurer la convergence globale de l'algorithme PES, la condition de passivité suivante doit être satisfaite

$$\operatorname{Re} \left\{ \frac{D(j\omega)}{A(j\omega)} \right\} > 0 \quad \forall \omega \quad (16)$$

Notons que cette condition dépend du polynôme dénominateur de la fonction de transfert du système réel qui est, par définition, inconnu.

Considérons le diagramme de Nyquist du transfert $\frac{D(j\omega)}{A(j\omega)}$, la condition (16) est respectée si le tracé n'apparaît pas dans le demi-plan gauche. En d'autres termes, l'argument de la fonction complexe $\frac{D(j\omega)}{A(j\omega)}$ doit être en valeur absolue inférieur à $\frac{\pi}{2}$. C'est le point clé de cette communication. Pour un système du premier ordre, et donc un filtre stationnaire également du premier ordre, la condition (16) est toujours respectée, i.e. la convergence globale est toujours obtenue.

III. DÉCOMPOSITION EN ÉLÉMENTS DU PREMIER ORDRE

L'idée clé de l'approche proposée est de décomposer un système d'ordre n en une somme de n sous-systèmes découplés du premier ordre. Cette décomposition est réalisée par le passage d'une représentation d'état canonique à une représentation diagonale.

A. Principe

Hypothèse 1:

- i- On dispose des mesures $\{u(t), y(t)\}$.
- ii- On dispose d'un modèle $\hat{H}(s, \tilde{\theta})$ avec une connaissance a priori sur les paramètres $\tilde{\theta}$.
- iii- Le système est constitué de pôles réels.

Notons que les deux premières hypothèses sont classiques pour les algorithmes ES.

Considérons le système d'ordre n défini par (1). Ce système peut être mis sous la forme d'une représentation d'état canonique

$$\begin{aligned} \dot{X} &= AX + Bu \\ Y &= CX \end{aligned} \quad (17)$$

où

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{bmatrix} \quad (18)$$

$$B = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \quad C = [b_0 \quad \dots \quad b_m \quad 0 \quad \dots \quad 0]$$

L'algorithme ci-dessous permet la décomposition du système en éléments du premier ordre par l'utilisation de la matrice de passage à une représentation d'état diagonale. Puis, l'identification de chaque sous-système est réalisée séparément pour satisfaire la condition de convergence globale (16).

Algorithme 1:

1. A partir des mesures $\{y(t)\}$, on approxime $\{\hat{y}(t), \dots, \frac{d^{n-1}\hat{y}}{dt^{n-1}}\}$ en utilisant une méthode de lissage par moindres carrés (voir Annexe).
2. A l'aide de $\{\hat{y}(t), \dots, \frac{d^{n-1}\hat{y}}{dt^{n-1}}\}$, $u(t)$, $\tilde{\theta}$ et de la représentation canonique (17), on calcule le vecteur $\hat{X} = [\hat{x}_1(t), \dots, \hat{x}_n(t)]^T$.
3. On calcule les valeurs propres $\tilde{\lambda}_i$ pour $i = 1, \dots, n$ de (17) en considérant les paramètres connus a priori $\tilde{\theta}$.
4. On détermine la matrice \tilde{P} de changement de base permettant d'obtenir une représentation d'état diagonale.
5. On détermine les signaux $Z = [z_1(t), \dots, z_n(t)]^T$ à l'aide de

$$Z = \tilde{P}^{-1} \hat{X} \quad (19)$$

6. On obtient les signaux $Z^* = [z_1^*(t), \dots, z_n^*(t)]^T$ grâce à

$$Z^* = \tilde{C}^T \tilde{P} Z \quad (20)$$

$z_i^*(t)$ est la réponse à l'excitation $u(t)$ de la fonction de transfert du premier ordre

$$\frac{\beta_i}{\alpha_i + s} \quad (21)$$

7. On applique l'algorithme PES pour identifier les paramètres β_i et α_i de chaque élément du premier ordre.

8. Enfin, on calcule les paramètres $\hat{\theta}$ par simple calcul algébrique

$$\frac{\beta_1}{\alpha_1 + s} + \dots + \frac{\beta_n}{\alpha_n + s} = \frac{\hat{b}_0 + \dots + \hat{b}_m s^m}{\hat{a}_0 + \dots + \hat{a}_{n-1} s^{n-1} + s^n}$$

Illustrons cet algorithme à travers un exemple.

Exemple 1: Considérons un système du second ordre avec zéro

$$H(s) = \frac{b_0 + b_1 s}{a_0 + a_1 s + s^2} \quad (22)$$

représenté sur la figure 1, et qui peut être exprimé sous forme de représentation d'état canonique

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \quad (23)$$

$$y = \begin{bmatrix} b_0 & b_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

On dispose d'une connaissance a priori sur les paramètres $\tilde{\theta} = [\tilde{a}_0, \tilde{a}_1, \tilde{b}_0, \tilde{b}_1]^T$.

Après avoir réalisé l'étape 1, l'étape 2 est la solution $\hat{X} = [\hat{x}_1, \hat{x}_2]^T$ de

$$\begin{bmatrix} \tilde{b}_0 & \tilde{b}_1 \\ -\tilde{b}_1 \tilde{a}_0 & \tilde{b}_0 - \tilde{b}_1 \tilde{a}_0 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} \frac{d\hat{y}}{dt} \\ \hat{y} - \tilde{b}_1 u \end{bmatrix} \quad (24)$$

A l'étape 3, les valeurs propres $\tilde{\lambda}_1$ et $\tilde{\lambda}_2$ sont données par

$$\tilde{\lambda}_i = \frac{-\tilde{a}_1 \pm \sqrt{\tilde{a}_1^2 - 4\tilde{a}_0}}{2} \quad (25)$$

On peut choisir, comme matrice de passage à la forme diagonale, la matrice suivante

$$\tilde{P} = \begin{bmatrix} 1 & 1 \\ \tilde{\lambda}_1 & \tilde{\lambda}_2 \end{bmatrix} \quad (26)$$

L'étape 5 donne les états $Z = [z_1, z_2]^T$ de la représentation diagonale suivante

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} \tilde{\lambda}_1 & 0 \\ 0 & \tilde{\lambda}_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \frac{1}{\tilde{\lambda}_2 - \tilde{\lambda}_1} \begin{bmatrix} -1 \\ 1 \end{bmatrix} u \quad (27)$$

$$\hat{y} = \begin{bmatrix} \tilde{b}_0 + \tilde{b}_1 \tilde{\lambda}_1 & \tilde{b}_0 + \tilde{b}_1 \tilde{\lambda}_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

L'étape 6 fournit les signaux

$$\begin{aligned} z_1^* &= (\tilde{b}_0 + \tilde{b}_1 \tilde{\lambda}_1) z_1 \\ z_2^* &= (\tilde{b}_0 + \tilde{b}_1 \tilde{\lambda}_2) z_2 \end{aligned} \quad (28)$$

dont la somme est la reconstruction de la sortie \hat{y} comme le montre la figure 2.

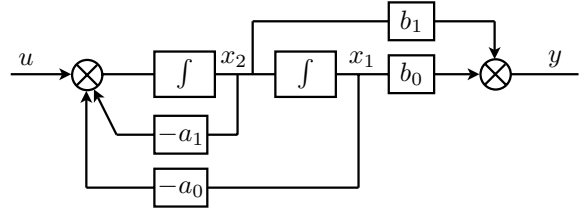


Fig. 1. Exemple 1 - Système d'ordre 2

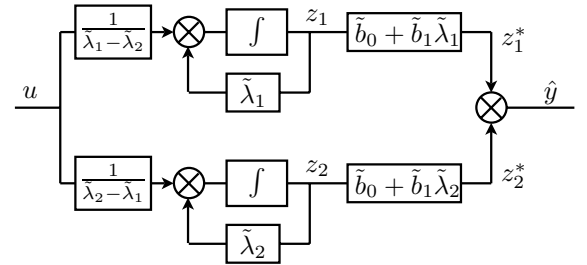


Fig. 2. Exemple 1 - Décomposition en 2 éléments du premier ordre

B. Discussion

En identifiant séparément chaque premier ordre, la condition de passivité (16) est respectée, et donc, la convergence globale est obtenue. On devine que l'étape de lissage par moindres carrés peut engendrer une perte d'information sur les données de sortie. De plus, la connaissance a priori sur les paramètres $\tilde{\theta}$ est généralement peu précise. D'où la présence d'une erreur d'estimation. Cependant, les paramètres obtenus sont voisins des paramètres réels, et peuvent donc être utilisés pour initialiser un algorithme à erreur de sortie.

Dans l'étape 1, pour approximer la sortie et ses dérivées, la méthode de lissage par moindres carrés présentée en annexe est utilisée. Elle consiste en une approximation sur une fenêtre glissante $[-K, K]$ de la sortie par un polynôme. Le principal avantage de cette approche est qu'elle ne nécessite pas de choix sur la dynamique d'un éventuel filtre ou observateur. Par contre, l'utilisateur doit faire le choix de la largeur $2K$ de la fenêtre et de l'ordre r du polynôme. Pour les applications de la section suivante, on a choisi $K = 15$ et $r = 4$.

L'étape 2 repose sur un calcul direct du vecteur d'état à partir de l'approximation de la sortie et de ses dérivées et des paramètres connus a priori. On remarque dans ce calcul (24) que la dynamique du modèle connu a priori n'intervient pas. L'approximation du vecteur d'état est assez fidèle au vecteur d'état réel, aux imprécisions près sur la connaissance a priori du gain et des paramètres \tilde{b}_i du modèle. Par contre, la dynamique du modèle connu a priori intervient dans les étapes

suivantes.

L'étape 4 nécessite l'inversion de la matrice \tilde{P} . Une restriction s'impose. Le modèle connu a priori $\hat{H}(s, \tilde{\theta})$ ne doit pas présenter de pôles réels doubles qui rendent la matrice \tilde{P} non inversible. Ce problème peut être facilement résolu ; si le besoin de choisir deux pôles identiques existe, il suffit de choisir deux pôles voisins.

Bien entendu, les développements ci-dessus ne permettent de considérer que des systèmes à pôles réels. L'identification de systèmes à pôles complexes n'a pas été considérée.

IV. SIMULATION ET RÉSULTATS

Pour illustrer l'approche proposée, deux systèmes, du second ordre et du troisième ordre, sont considérés.

A. Système du second ordre

Considérons le système suivant

$$H(s) = \frac{2}{s^2 + 3s + 1} \quad (29)$$

L'entrée est une Séquence Binaire Pseudo-Aléatoire (SBPA). Un bruit blanc est ajouté aux mesures de la sortie avec un rapport variance du signal de sortie sur variance du bruit (S/N) égal à 10. La figure 3 montre l'entrée, la sortie sans bruit et la sortie bruitée.

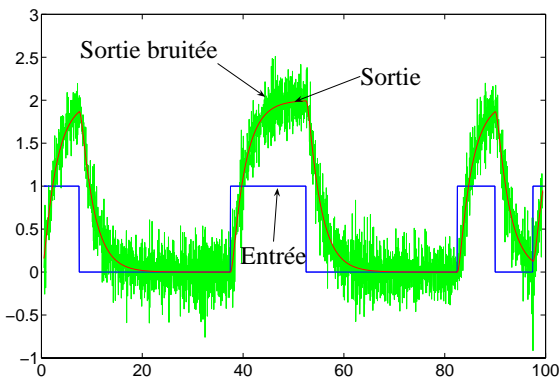


Fig. 3. Données d'entrée-sortie

Une fonction de transfert du second ordre exprimée en fonction des constantes de temps réelles τ_1 et τ_2 est définie par

$$H(s) = \frac{G/\tau_1\tau_2}{s^2 + (\tau_1 + \tau_2)/\tau_1\tau_2 s + 1/\tau_1\tau_2} \quad (30)$$

Pour le système (29) considéré, les constantes de temps sont $\tau_1 = 2.618s$ et $\tau_2 = 0.3820s$, et le gain est $G = 2$.

On considère un modèle qui présente la même structure que le système. L'objectif est d'identifier les paramètres b_0, a_0 et a_1 .

On utilise d'abord l'algorithme standard de Gauss-Newton initialisé avec les paramètres $\hat{b}_0 = 15, \hat{a}_0 = 5$ et $\hat{a}_1 = 10$, soient $\hat{\tau}_1 = 1.8943s, \hat{\tau}_2 = 0.1056s$ et $\hat{G} = 3$. Pour cette initialisation, l'algorithme de Gauss-Newton diverge.

On applique pour la même initialisation l'algorithme 1 proposé dans la section précédente. L'algorithme PES donne les deux sous-modèles du premier ordre suivants : $\hat{H}_1(s) = \frac{0.7392}{s+0.3432}$ et $\hat{H}_2(s) = \frac{-0.14}{s+1.287}$. La fonction de transfert du second ordre correspondante est

$$\hat{H}_{PES}(s) = \hat{H}_1(s) + \hat{H}_2(s) = \frac{0.5992s + 0.9033}{s^2 + 1.63s + 0.4417} \quad (31)$$

soient $\hat{\tau}_1 = 2.9138s, \hat{\tau}_2 = 0.777s$ et $\hat{G} = 2.0451$.

Le gain et la constante de temps principale sont assez proches des vraies valeurs. Cependant, la seconde constante de temps est le double de la valeur réelle. C'est probablement dû à l'existence d'un zéro au numérateur de la fonction de transfert résultante (31).

La figure 4 montre la sortie mesurée sans bruit, la sortie obtenue à partir des paramètres initiaux et la sortie obtenue à l'issue de l'algorithme 1. On remarque, qu'en terme d'entrée-sortie, le modèle obtenu a un bon comportement.

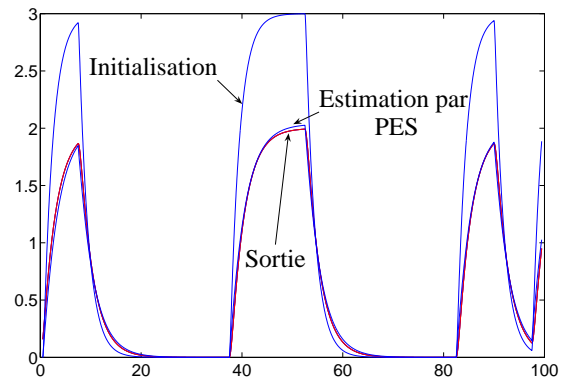


Fig. 4. Résultats obtenus avec l'algorithme 1

Les paramètres estimés avec cette approche sont maintenant utilisés pour initialiser l'algorithme standard de Gauss-Newton, afin d'éliminer les erreurs engendrées par le zéro au numérateur ou par le lissage utilisé dans la procédure. L'évolution du critère quadratique et des paramètres est présentée sur la figure 5. A partir de cette initialisation et après quatre itérations, l'algorithme de Gauss-Newton a convergé. Le modèle alors estimé est

$$\hat{H}_{ES}(s) = \frac{2.5986}{s^2 + 3.8025s + 1.2854} \quad (32)$$

soient $\hat{\tau}_1 = 2.6665s, \hat{\tau}_2 = 0.2918s$ et $\hat{G} = 2.0216$.

Il est à noter que l'algorithme de Gauss-Newton a été utilisé pour identifier les paramètres des sous-modèles du premier ordre avec l'initialisation de départ. Mais, de nouveau, une

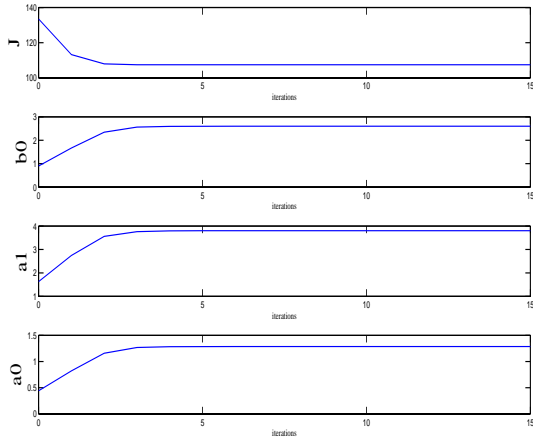


Fig. 5. Évolution du critère et des paramètres

divergence a été constatée prouvant ainsi l'intérêt de l'algorithme PES.

La figure 6 montre le résultat obtenu avec l'algorithme de Gauss-Newton initialisé avec les paramètres estimés avec l'algorithme PES. La sortie du modèle se superpose à la sortie mesurée.

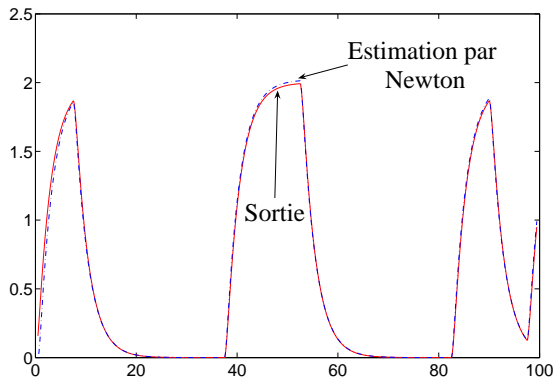


Fig. 6. Résultats obtenus avec l'algorithme de Gauss-Newton initialisé par l'algorithme PES

La table I résume les paramètres obtenus aux différentes étapes de l'identification du système du second ordre.

Pour conclure sur l'étude de ce système du second ordre, on peut noter sur la figure 6 un bon comportement du modèle $\hat{H}_{ES}(s)$ estimé. Dans la table I, on constate des erreurs relativement importantes sur les paramètres. Celles-ci peuvent s'expliquer par le niveau de bruit, d'une part, et d'autre part, par la difficulté d'identifier un système ayant des constantes de temps éloignées. Nous avons retenu ce système car il permet de mettre en défaut l'algorithme de Gauss-Newton et de montrer les propriétés de notre approche.

 TABLE I
RÉSULTATS POUR LE SYSTÈME DU SECOND ORDRE

	Système réel	Paramètres initiaux	Modèle PES	Modèle ES
a_0	1	5	0.4417	1.2854
a_1	3	10	1.63	3.8025
b_0	2	15	0.9033	2.5986
b_1			0.5992	
τ_1	2.618	1.8943	2.9138	2.6665
τ_2	0.3820	0.1056	0.777	0.2918
G	2	3	2.0451	2.0216

B. Système du troisième ordre

Considérons le système du troisième ordre suivant

$$H(s) = \frac{1}{s^3 + 3s^2 + 2.99s + 0.99} \quad (33)$$

L'entrée est une SBPA. Un bruit blanc est ajouté aux mesures de la sortie avec un rapport S/N égal à 100.

L'étude de ce système suit les mêmes étapes que pour le système du second ordre. L'objectif des simulations sur un système du troisième ordre est de montrer la capacité de notre approche à estimer 3 modes distincts.

La figure 7 montre les données d'entrée et de sortie. La sortie obtenue à partir des paramètres initiaux et la sortie obtenue à l'issue de l'algorithme 1 sont présentées sur la figure 8.

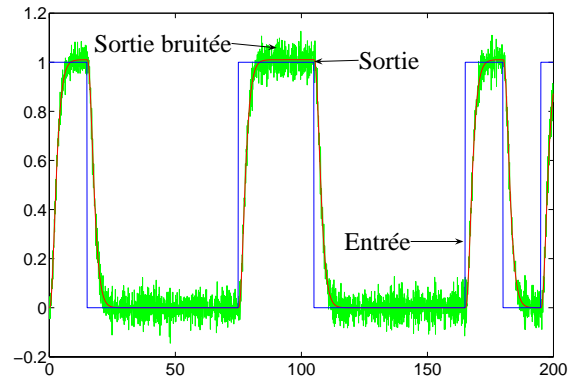


Fig. 7. Données d'entrée-sortie

L'algorithme de Gauss-Newton est ensuite initialisé avec les paramètres obtenus avec l'algorithme PES. La figure 9 montre le résultat alors obtenu.

La table II résume les paramètres obtenus aux différentes étapes de l'identification du système du troisième ordre.

L'étude de ce système du troisième ordre permet de montrer le bon comportement de l'approche. Les paramètres estimés à l'issue de l'algorithme ES initialisé par le modèle PES sont proches des paramètres réels. L'algorithme ES initialisé

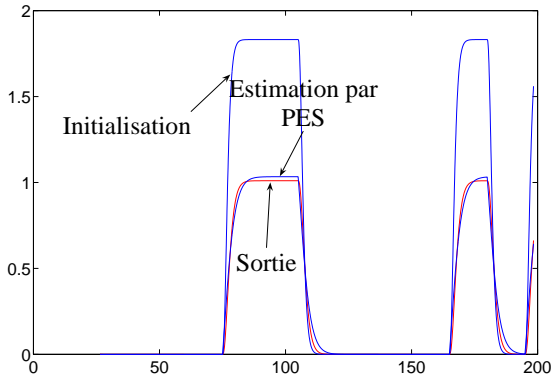


Fig. 8. Résultats obtenus avec l'algorithme 1

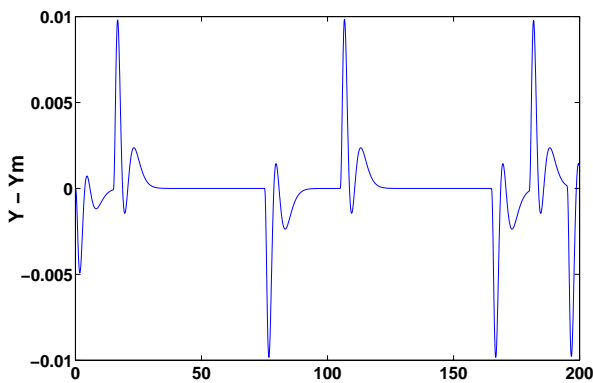


Fig. 9. Écart entre la sortie réel et la sortie du modèle ES

directement avec les paramètres initiaux ne permet pas la convergence vers les paramètres réels.

V. CONCLUSION

Dans cette communication, une nouvelle approche pour l'initialisation des algorithmes à erreur de sortie est présentée. Elle implique une modification de l'algorithme de Gauss-Newton pour améliorer la convergence. Cette modification consiste à introduire un filtre stationnaire dans le calcul des fonctions de sensibilité. Ces fonctions ne sont donc pas les vraies fonctions de sensibilité, et sont donc appelées pseudo-fonctions de sensibilité. L'algorithme associé est alors nommé algorithme à Pseudo-Erreur de Sortie.

TABLE II

RÉSULTATS POUR LE SYSTÈME DU TROISIÈME ORDRE

	Système réel	Paramètres initiaux	Modèle PES	Modèle ES
a_0	0.99	2.73	0.2244	0.9696
a_1	2.99	5.87	1.108	2.9295
a_2	3	4.2	1.824	2.9397
b_0	1	5	0.2246	0.9787
b_1			0.3875	
b_2			0.1804	

L'étude théorique de la convergence de l'algorithme PES a permis d'établir une condition de passivité assurant la convergence globale. Cette condition est systématiquement assurée dans le cas d'un système et d'un filtre stationnaire du premier ordre. Pour bénéficier de cette propriété, nous proposons une décomposition du système en éléments du premier ordre avant l'identification. Le modèle obtenu est alors utilisé pour initialiser un algorithme à erreur de sortie.

Une perspective immédiate est d'étendre cette approche au cas des systèmes présentant des pôles complexes.

ANNEXE

Dans l'étape 1 de l'algorithme 1, la méthode de lissage par moindres carrés est utilisée pour estimer la sortie et ses dérivées $\hat{y}, \frac{d\hat{y}}{dt}, \dots, \frac{d^r \hat{y}}{dt^r}$.

Soit $\{y(t)\}$ un ensemble de N mesures de la sortie bruitée aux instants $t = kt_s$ où t_s est la période d'échantillonnage.

Considérons le polynôme suivant $f(\mu) = f_0 + f_1\mu + \dots + f_r\mu^r$ où $\mu = t - kt_s$.

$f(\mu)$ est ajusté sur les données $y(\mu)$ par minimisation du critère quadratique

$$J = \sum_{k=-K}^K (y(t + kt_s) - f(t + kt_s))^2 \quad (34)$$

Les paramètres \hat{f}_i sont alors obtenus par les moindres carrés.

Les dérivées successives de $f(\mu)$ sont données par

$$\begin{aligned} f(\mu) &= \hat{f}_0 + \hat{f}_1\mu + \dots + \hat{f}_r\mu^r \\ \frac{d}{d\mu} f(\mu) &= \hat{f}_1 + 2\hat{f}_2\mu + \dots + r\hat{f}_r\mu^{r-1} \\ \frac{d^2}{d\mu^2} f(\mu) &= 2\hat{f}_2 + 6\hat{f}_3\mu + \dots + r(r-1)\hat{f}_r\mu^{r-2} \\ &\dots \end{aligned} \quad (35)$$

Ainsi, on obtient une approximation de la sortie et de ses dérivées aux instants $t = kt_s$ avec $k = K, \dots, N - K$ grâce à

$$\begin{aligned} \hat{y}(kt_s) &= \hat{f}_0 \\ \frac{d}{dt} \hat{y}(kt_s) &= \hat{f}_1 \\ \frac{d^2}{dt^2} \hat{y}(kt_s) &= 2\hat{f}_2 \\ &\dots \end{aligned} \quad (36)$$

RÉFÉRENCES

- [1] Jacques Richalet, A. Rault, et R. Pouliquen. *Identification des processus par la méthode du modèle*. Gordon and Breach, 1971.
- [2] I.D. Landau. *Adaptive control : the model reference approach*. Marcel Dekker, Control and systems theory, Vol. 8, 1979.
- [3] L. Pronzato et E. Walter. Eliminating suboptimal local minimizers in nonlinear parameter estimation. *Technometrics*, 43(4) :434-442, 2001.
- [4] L. Pronzato et E. Walter. Comment éviter les optimiseurs locaux parasites en estimation paramétrique non linéaire. *APII-JESA*, 36(3) :351-362, 2002.
- [5] M. Djamai, E. Tohme, R. Ouvrard, et S. Bachir. Continuous-time model identification using reinitialized partial moments - application to power amplifier modeling. *IFAC Symposium on System Identification*, Newcastle, Australia, 2006.

- [6] E. Tohme, R. Ouvrard, J.C. Trigeassou, A. Abche, et T. Poinot. Initialisation des algorithmes d'optimisation - comparaison de deux approches à erreur d'équation. *Soumis aux Journées Identification et Modélisation Expérimentale*, Poitiers, France, 2006.
- [7] I.D. Landau. Unbiased recursive identification using model reference adaptive techniques. *IEEE Trans. Automat. Contr.*, AC-21(2) :194–202, April 1976.
- [8] J.C. Trigeassou, F.J. Carillo, T. Poinot, et O. Grospeaud. Convergence des algorithmes à erreur de sortie hors-ligne et récurrents. *APII - Journal Européen des Systèmes Automatisés*, 36 :397–415, 2002.
- [9] J.C. Trigeassou, F.J. Carillo, T. Poinot, et O. Grospeaud. Convergence des algorithmes à erreur de sortie hors-ligne et récurrents. *Journées Identification et Modélisation Expérimentale*, pages 198–207, Nancy, France, 2001.
- [10] E. Tohme, R. Ouvrard, J.C. Trigeassou, et A. Abche. A pseudo-output error algorithm to improve global convergence. *The 32nd Annual Conference of the IEEE Industrial Electronics Society IECON'06*, Paris, France, 2006.
- [11] J.C. Trigeassou. *Contribution à l'extension de la méthode des moments en automatique. Application à l'identification des systèmes linéaires*. Thèse d'état, Université de Poitiers, France, 1987.
- [12] L. Ljung. On positive real transfer functions and the convergence of some recursive schemes. *IEEE Trans. Automat. Contr.*, AC-22(4) :539–551, August 1977.
- [13] L. Ljung. Analysis of recursive stochastic algorithms. *IEEE Trans. Automat. Contr.*, AC-22(4) :551–575, August 1977.