RÉFÉRENCES BIBLIOGRAPHIQUES

[ABU88] ABU EL ATA S. & COIC A. (1988) " Commande prédictive par inversion : application aux systèmes non linéaires ", Contrat DRET n°87/1226 du 6 avril 1988, ADERSA.

[AST84] ASTRÖM K. J. & WITTENMARK B. (1984) Computer controlled Systems, Prentice Hall, Englewood Cliffs (New Jersey).

[AYE81] AYELES D. (1981) " Generic observability of differentiable systems ", SIAM Journal of Control and Optimization, Vol.5 No.19, pp. 595-603.

[BAR93] BARRON A. (1991) " Universal approximation bounds for superposition of a sigmoidal function ", IEEE Trans. Information theory IT-39, pp. 930-945.

[BEN94] BENVENISTE A., JUDITSKY A., DELYON B., ZHANG Q. & GLORENNEC P.-Y. (1994) " Wavelets in identification ", Preprints of the 10th IFAC Symposium on Identification, Copenhague, 4-6 juillet 1994.

[BIL92] BILLINGS S. & CHEN S. (1992) " Neural networks and system identification ", in [HUS92], pp. 181-205.

[BLO] MM. BLONDELEAU, SEZNEC & PIERRE. Architecture et dynamique automobile, Service Formation de la DRD Renault.

[BOR90] BORNE P., DAUPHIN-TANGUY G., RICHARD J. P., F. ROTELLA & ZAMBETTAKIS I. (1990) Commande et optimisation des processus, Éditions Technip, Paris.

[BRO83] BROCKETT R. W. (1983) " Asymptotic stability and feedback stabilization ", in Differential geometric control theory, Proceedings of the conference held at Michigan Technological University, June 28-July 2, 1982, Progress in Mathematics Vol.27, Birkhäuser, pp. 181-191.

[CAN91] CANUDAS DE WIT C. & SORDALEN O. J. (1991) " Exponential stabilization of mobile robots with nonholonomic constraints ", IEEE Conference on Decision and Control, pp. 692-697.

[CHA90] CHANTRE P., CANAS D. & DARDENNES J. (1990) " Control by inversion of non-linear non-minimum-phase systems ", Proceedings of the IASTED'90, Autriche, pp. 1-6.

[CHE89] CHEN S. & BILLINGS (1989) " Recursive prediction error parameter estimation for non linear models ", Int. J. Control Vol.49 No.2, pp. 569-594.

[CHE90a] CHEN S., BILLINGS S. A. & GRANT P. M. (1990) " Non linear system identification using neural networks ", Int. J. Control Vol.51 No.6, pp. 1191-1214.

[CHE90b] CHEN S., COWAN C.F.N., BILLINGS S. A. & GRANT P. M. (1990) " Parallel recursive prediction error algorithm for training neural networks ", Int. J. Control Vol.51 No.6, pp. 1215-1228.

[CYB89] CYBENKO G. (1989) " Approximation by superpositions of a sigmoïdal function ", Mathematics of Control, Signals and Systems 2, pp. 303-314.

[ETA94] Analyse des essais de guidage-pilotage du véhicule REMI, ETAS 24-30 mars 1994, document Confidentiel Industrie SAGEM SE 96/22/94.

[FAR93] FARGEON C. & QUIN J.-P. (1993) Robotique mobile, Teknea.

[FRA93] FRAPPIER G. (1993) " MINERVE : navigation - guidage - pilotage de véhicules autonomes ", Journée thématique DRET : " Vers une plus grande autonomie des robots mobiles ", janvier 1993, Paris.

[FUN89] FUNAHASHI K. (1989) " On the approximate realization of continous mappings by neural networks ", Neural Networks Vol.2, pp. 183-192.

[GOO84] GOODWIN G. C. & SIN K. S. (1984) Adaptive filtering prediction and control, Prentice-Hall, New Jersey.

[GRO94] GRONDIN B. (1994) " Les réseaux de neurones pour la modélisation et la conduite des réacteurs chimiques : simulations et expérimentations ", Thèse de doctorat de l'Université de Bordeaux I.

[HED91] HEDRICK J. K., McMAHON D., NARENDRAN V. & SWAROOP D. (1991) " Longitudinal vehicle controller design for IVHS systems ", Proceedings of 1991 American Control Conference, Boston, MA, June 1991, pp. 3107-3112.

[HOR89] HORNIK K., STINCHCOMBE M. & WHITE H. (1989) " Multilayer feedforward networks are universal approximators ", Neural Networks 2, pp. 359-366.

[HUN92] HUNT K. J . & SBARBARO D. (1992) " Studies in neural network based control ", in [HUS92], pp. 94-122.

[HUS92] HUNT K. J . & SBARBARO D. (1992) Neural networks for control and systems, Peter Peregrinus Ltd., London.

[JOR85] JORDAN M. I. (1985) " The learning of representations for sequential performance ", Doctoral Dissertation, University of California, San Diego.

[JUR93] JURIE F., RIVES P., GALLICE J. & BRAME J. L. (1993) " High speed vehicle guidance based on vision ", Preprints of the 1st IFAC International Workshop on Intelligent Autonomous Vehicles, pp. 205-210.

[KOS92] KOSKO B. (1992) Neural networks and fuzzy systems, Prentice Hall, Englewood Cliffs, New Jersey.

[LAN93] LANDAU I. D. (1993) Identification et commande des systèmes, 2ème édition, Hermès, Paris.

[LAR89] LARMINAT P. de (1989) " La commande robuste : un tour d'horizon ", Journées nationales du G. R. " Automatique ", 26-28 avril 1989, Biarritz.

[LAR93] LARMINAT P. de (1993) Automatique ; Commande des systèmes linéaires, Hermes.

[LAT91] LATOMBE J. C. (1991) Robot motion planning, Kluwer Academic Publishers, Boston.

[LEO85a] LEONTARITIS I. J. & BILLINGS S. A. (1985) " Input-output parametric models for non-linear systems. Part I : deterministic non-linear systems ", Int. J. Control Vol.41 No.2, pp. 303-328.

[LEO85b] LEONTARITIS I. J. & BILLINGS S. A. (1985) " Input-output parametric models for non-linear systems. Part II : stochastic non-linear systems ", Int. J. Control Vol.41 No.2, pp. 329-344.

[LEO87] LEONTARITIS I. J. & BILLINGS S. A. (1987) " Model selection and validation methods for non-linear systems ", Int. J. Control Vol.45 No.1, pp. 311-341.

[LEV92] LEVIN A. U. (1992) " Neural networks in dynamical systems ; a system theoretic approach ", PhD Thesis, Yale University.

[LEV93] LEVIN A. U. & NARENDRA K. S. (1993) " Control of nonlinear dynamical systems using neural networks : Controllability and stabilization ", IEEE Trans. on Neural Networks Vol.4 N°2, pp. 1011-1020.

[LIU93] LIUBAKKA M. K., RHODE D. S., WINKELMAN J. R. & KOKOTOVIC P. V. (1993) " Adaptive automotive speed control ", IEEE Trans. on Robotics and Automation Vol.38 No.7, pp. 1-9.

[LJU87] LJUNG L. (1987) System identification ; theory for the user, Prentice Hall, Englewood Cliffs, New Jersey.

[LJU91] LJUNG L. (1991) " Issues in system identification ", IEEE Control Systems Magazine Vol.11 N°1, pp. 25-29.

[MCU43] McCULLOCH W. & PITTS W. (1943) " A logical calculus of the ideas immanent in the brain ", Bull. Math. Biophysics No.5.

[MIL91] MILLER W. T., SUTTON R. S. & WERBOS P. J. (1991) Neural networks for control, MIT Press, Cambridge MA.

[MIN83] MINOUX M. (1983) Programmation mathématique ; théorie et algorithmes, Dunod.

[MIN93a] MINERVE.3 (1993) Contrôle de la direction. Rapport intermédiaire. Contrat DRET N°91.426. Mars 1993.

[MIN93b] MINERVE.3 (1993) Contrôle de la vitesse. Rapport intermédiaire. Contrat DRET N°91.426. Octobre 1993.

[MIN94] MINERVE.3 (1994) Rapport de synthèse. Contrat DRET N°91.426. Février 1994.

[MOR89] MORARI M. & ZAFIRIOU E. (1989) Robust process control, Prentice-Hall International Editions.

[MOR93] MORARI M. (1993) " Some control problems in the process industries ", in [TRE93], pp. 55-78.

[NAR90] NARENDRA K. S. & PARTHASARATHY K. (1990) " Identification and control of dynamical systems using neural networks ", IEEE Trans. on Neural Networks vol.1 No.1, pp. 4-27.

[NAR91] NARENDRA K. S. & PARTHASARATHY K. (1991) " Gradient methods for the optimization of dynamical systems containing neural networks ", IEEE Trans. on Neural Networks Vol.2 No.2, pp. 252-262.

[NAR92] NARENDRA K. S. (1992) " Adaptive control of dynamical systems using neural networks ", in Handbook of intelligent control : neural, fuzzy and adaptive approaches, D. A. White & D. A. Sofge eds., Van Nostrand Reinhold, New York.

[NAS90] NASH J. C. (1990) Compact numerical methods for computers : linear algebra and function minimisation, Adam Hilger.

[NER92a] NERRAND O., ROUSSEL-RAGOT P., PERSONNAZ L. & DREYFUS G. (1992) “ Neural networks and non-linear adaptive filtering : unifying concepts and new algorithms ”, Neural Computation Vol.5 No.2, pp. 165-199.

[NER92b] NERRAND O. (1992) “ Réseaux de neurones pour le filtrage adaptatif, l'identification et la commande de processus ”, Thèse de doctorat de l'Université Paris VI.

[NER93] NERRAND O, PERSONNAZ L. & DREYFUS G. (1993) "Non-linear recursive identification and control by neural networks : a general framework", European Control Conference 1993.

[NOR88] NORTON J. P. (1988) An introduction to identification, Academic Press, London.

[PEN92] PENG H., HESSGURG T., TOMIZUKA M., ZHANG W., LIN Y., DEVLIN P. & SHLADOVER S. E. (1992) “ A theoretical and experimental study on vehicle lateral control ”, Proceedings of the 1992 American Control Conference, Chicago, June 1992, pp. 1738-1742.

[PLO94] PLOIX J.-L., DREYFUS G., CORRIOU J.-P., PASCAL D. (1994) “ From knowledge-based models to recurrent networks : an application to an industrial distillation process ”, NeuroNîmes’94.

[PLU94] PLUMER E. S. (1994) “ Optimal control of terminal processes using neural networks ”, soumis à IEEE Transactions on Neural Networks.

[POI02] POINCARÉ H. (1902) La science et l’hypothèse, Flammarion.

[POM91] POMERLEAU D. A. (1991) “ Efficient training of artificial neural networks for autonomous navigation ”, Neural ComputationVol.1 No.3, pp. 88-97.

[POM92] POMET J.-B., THUILOT B., BASTIN G. & CAMPION G. (1992) “ A hybrid strategy for the feedback stabilization of nonholonomic mobile robots ”, Proc. of the 1992 International Conference on Robotics and Automation, Nice, France, mai 1992, pp. 129-134.

[POM94] POMERLEAU D. A. (1994) “ Reliability estimation for neural network based autonomous driving ”, Robotics and Autonomous Systems 12, pp. 113-119.

[PSI91] PSICHOGIOS D. C. & UNGAR L. H. (1991) “ Direct and indirect model based control using artificial networks ”, Ind. Eng. Chem. Res. Vol.30 N°12, pp. 2564-2573.

[RIC91] RICHALET J. (1991) “ Model based predictive control ”, Cours de l’École supérieure d’Électricité.

[RIN93] RINTANEN K. T. (1993) “ Curvature-optimal path planning and servoing for autonomous vehicles : a neural net implementation ”, 1st IFAC International Workshop on Intelligent Autonomous Vehicles, University of Southampton, 18-21 april 1993.

[RIV93a] RIVALS I. (1993) “ Application des réseaux de neurones au pilotage automatique de véhicules ”, document Confidentiel Industrie SAGEM SE 96/01/93.

[RIV93b] RIVALS I., PERSONNAZ L., DREYFUS G. & CANAS D. (1993) " Real-time control of an autonomous vehicle : a neural network approach to the path-following problem ", 5th International Conference on Neural Networks and their Applications, pp. 219-229 (NeuroNîmes'93).

[RIV94] RIVALS I., CANAS D., PERSONNAZ L. & DREYFUS G. (1994) " Modeling and control of mobile robots and intelligent vehicles by neural networks ", IEEE Conference on Intelligent Vehicles, 24-26 octobre 1994, Paris, pp. 137-142.

[RUM86] RUMELHART D. E., HINTON G. E. & WILLIAMS R. J. (1986) " Learning internal representations by error back-propagation ", In Parallel Distributed Processing : explorations in the microstructure of cognition. Vol.1 : Foundations, D. E. Rumelhart, J. L. McLelland and the PDP Research Group eds., MIT Press, Cambridge MA, pp. 318-362.

[SAM90] SAMSON C. & AIT-ABDERRAHIM K. (1990) " Feedback control of a non-holonomic wheeled cart in cartesian space ", rapport de recherche INRIA N°1288, octobre 1990. Cité comme : IEEE Int. Conf. on Robotics and Automation, Sacramento, California, April 1991.

[SAM91a] SAMSON C. & AIT-ABDERRAHIM K. (1991) " Feedback stabilization of a non-holonomic wheeled mobile robot ", Int. Workshop on Intelligent Robots and Systems (IROS'91), Osaka, Japan.

[SAM91b] SAMSON C. (1991) " Velocity and torque feedback control of a non-holonomic cart ", in Advanced Robot Control, Proc. of the Int. Workshop on non-linear and adaptive control : Issues in Robotics, Grenoble, Nov. 21-23, 1990, vol.162, C. Canudas de Wit ed., Springer-Verlag.

[SAM91c] SAMSON C. (1991) " Time-varying feedback stabilization of non-holonomic car-like mobile robots ", rapport de recherche INRIA N°1515, Septembre 1991.

[SAM92] SAMSON C. (1992) " Path following and time-varying feedback stabilisation of a wheeled mobile robot ", Proc. Conf. ICARCV'92, Singapore, September 1992.

[SBA93] SBARBARO-HOFER D., NEUMERKEL D. & HUNT K. (1993) " Neural control of a steel rolling mill ", IEEE Control Systems Vol.13 No.3, June 1993, pp. 69-75.

[SCO92] SCOTT G. M., SHAVLIK J. W. & RAY W. H. (1992) " Refining PID controllers ", Neural Computation 4, pp. 746-757.

[SHI90] SHIN D. H. & SINGH S. (1990) " Vehicle and path models for autonomous navigation ", in Vision and navigation : the Carnegie Mellon Navlab, Thorpe C. E. ed., Kluwer Academic Publishers, Boston/Dordrecht/London, pp. 283-307.

[SHL78] SHLADOVER S. E. (1978) " Longitudinal control of automated guideway transit vehicle within platoons ", Journal of dynamic systems, measurement and control, Vol.100, December 1978, pp. 302-310.

[SJÖ93] SJÖBERG J. (1993) " Regularization issues in neural network models of dynamical systems ", LiU-TEK-LIC-1993:08, ISBN 91-7871-072-3, ISSN 0280-7971.

[SJÖ94] SJÖBERG J., HJALMARSSON H. & LJUNG L. (1994) " Neural networks in system identification ", Report LiTH-ISY-R-1622.

[SLO93] SLOTINE J.-J. E. & SANNER R. M. (1993) " Neural networks for adaptive control and recursive identification ", in [TRE93], pp. 381-436.

[SON93] SONTAG E. D. (1993) " Neural networks for control ", in [TRE93], pp. 339-380.

[SOR92] SORDALEN O. J. & CANUDAS DE WIT C. (1992) " Evaluation of an exponential control law for a mobile robot : from regulation to tracking ", Proceedings of the 1992 IEEE International Conference on Robotics and Automation, Nice, France, mai 1992.

[TRE93] TRENTELMAN H. L. & WILLEMS J. C. eds. (1993) Essays on control : perspectives in the theory and its applications, Birkhäuser , Boston.

[URB94] URBANI D., ROUSSEL-RAGOT P., PERSONNAZ L. & DREYFUS G. (1994) " The selection of neural models of non-linear dynamical systems by statistical tests ", Neural Networks for Signal Processing, Proceedings of the 1994 IEEE Workshop, pp. 229-237.

[VDB93] Van den BOGAERT T., LEMOINE P., VACHERAND F. & DO S. (1993) " Obstacle avoidance in PANORAMA ESPRIT II project ", 1st IFAC International Workshop on Intelligent Autonomous Vehicles, pp. 48-53.

[VDM93] VAN DER MOLEN G. M. (1993) " Modelling and control of a wheeled mobile robot ", Preprints of the 1st IFAC International Workshop on Intelligent Autonomous Vehicles, pp. 289-294.

[WAI89] WAIBEL A., HANAZAWA T., HINTON G., SHIKANO K. & LANG K. (1989) " Phoneme recognition using Time-Delay Neural Networks ", IEEE Trans. on Acoustics, Speech, and Signal Processing Vol.37, pp. 328-339.

[WHI92] WHITE D. A. & JORDAN M. I. (1992) " Optimal control : a foundation for intelligent control ", in [WHS92], pp. 185-214.

[WHS92] WHITE D. A. & SOFGE D. A. (1992) Handbook of intelligent control : neural, fuzzy and adaptive approaches, Van Nostrand Reinhold, New York.

Annexe I

APPRENTISSAGE DES RÉSEAUX DE NEURONES

Position du problème.

Nous considérons ici le problème pratique de l'apprentissage d'un réseau de neurones bouclé décrit par la représentation d'état générale (forme canonique, chapitre 1 §I.2) :

$$\begin{cases} S(k+1) = \varphi_{RN}\left(S(k),\, I(k);\, C\right) \\ Y(k) = \psi_{RN}\left(S(k),\, I(k);\, C\right) \end{cases}$$

où nous notons $I(k) \in \mathbb{R}^{NI}$ le vecteur des entrées externes du réseau à l'instant k, $S(k) \in \mathbb{R}^{NS}$ le vecteur des variables d'état du réseau à l'instant k, $S(k+1) \in \mathbb{R}^{NS}$ le vecteur des variables d'état du réseau à l'instant k+1, $Y(k) \in \mathbb{R}^{NY}$ le vecteur des sorties du réseau à l'instant k, et C les coefficients du réseau. φ_{RN} (. , . ; C) et ψ_{RN} (. , . ; C) représentent les fonctions réalisées par le réseau de neurones de la forme canonique interconnectés avec les coefficients C.

La tâche du réseau est définie par (chapitre 1 §III.2) :

- des séquences d'apprentissage constituées d'une séquence appliquée aux entrées externes {I(k)}, et une séquence de valeurs désirées correspondantes {D(k)} pour les sorties du réseau ;
- une fonction de coût, définie à l'itération i sur une fenêtre fixe englobant toute la longueur N de la séquence d'apprentissage :

$$J(C, i) = \frac{1}{2} \sum_{k=1}^{N} E^i(k)^T W\, E^i(k) = \frac{1}{2} \sum_{k=1}^{N} \left(D(k) - Y^i(k)\right)^T W \left(D(k) - Y^i(k)\right)$$

où C représente les coefficients du réseau C(i-1) disponibles à l'itération i, $E^i(k)$ le vecteur des erreurs à l'instant k et à l'itération i, W une matrice définie positive (souvent diagonale), D(k) le vecteur des sorties désirées à l'instant k, et $Y^i(k)$ le vecteur des sorties du réseau à l'instant k et à l'itération i.

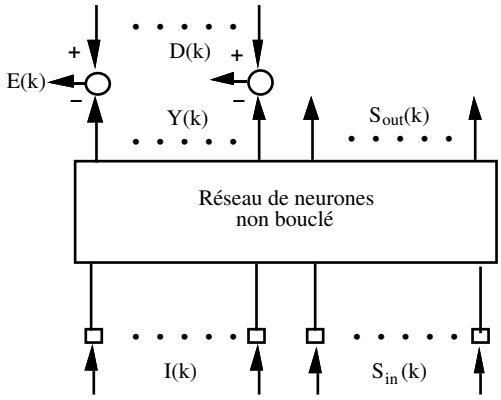

Figure 1.
Copie n°k utilisée pour l'apprentissage.

Notations.

Si l'on développe " spatialement " le comportement temporel du réseau [NER92b], on obtient un grand réseau non bouclé, dans lequel le réseau non bouclé de la forme canonique intervient N fois en cascade avec les coefficients disponibles à l'itération i. Les multiples interventions de ce même réseau sont appelées " *copies* ". Seules les valeurs des entrées et des sorties des neurones sont différentes d'une copie à l'autre. La copie numéro k est représentée sur la figure 1 (en omettant les indices i se référant à l'itération i).

Les entrées de la copie k sont :

- les entrées externes à l'instant k, I(k) $\mathbb{R}^{NI}$ (séquence d'apprentissage),
- les variables d'état d'entrée à l'instant k, $S^{in}(k) \in \mathbb{R}^{NS}$, avec $S^{in}(k)=S^{out}(k-1)$.

Les sorties de la copie k sont :

- les sorties à l'instant k, $Y(k) \in \mathbb{R}^{NY}$, qui sont les activités de NY neurones de sortie,
- les variables d'état de sortie à l'instant k, $S^{out}(k) \in \mathbb{R}^{NS}$, qui sont les activités de NS neurones d'état.

$D(k) \in \mathbb{R}^{NY}$ est le vecteur des sorties désirées à l'instant k (séquence d'apprentissage).

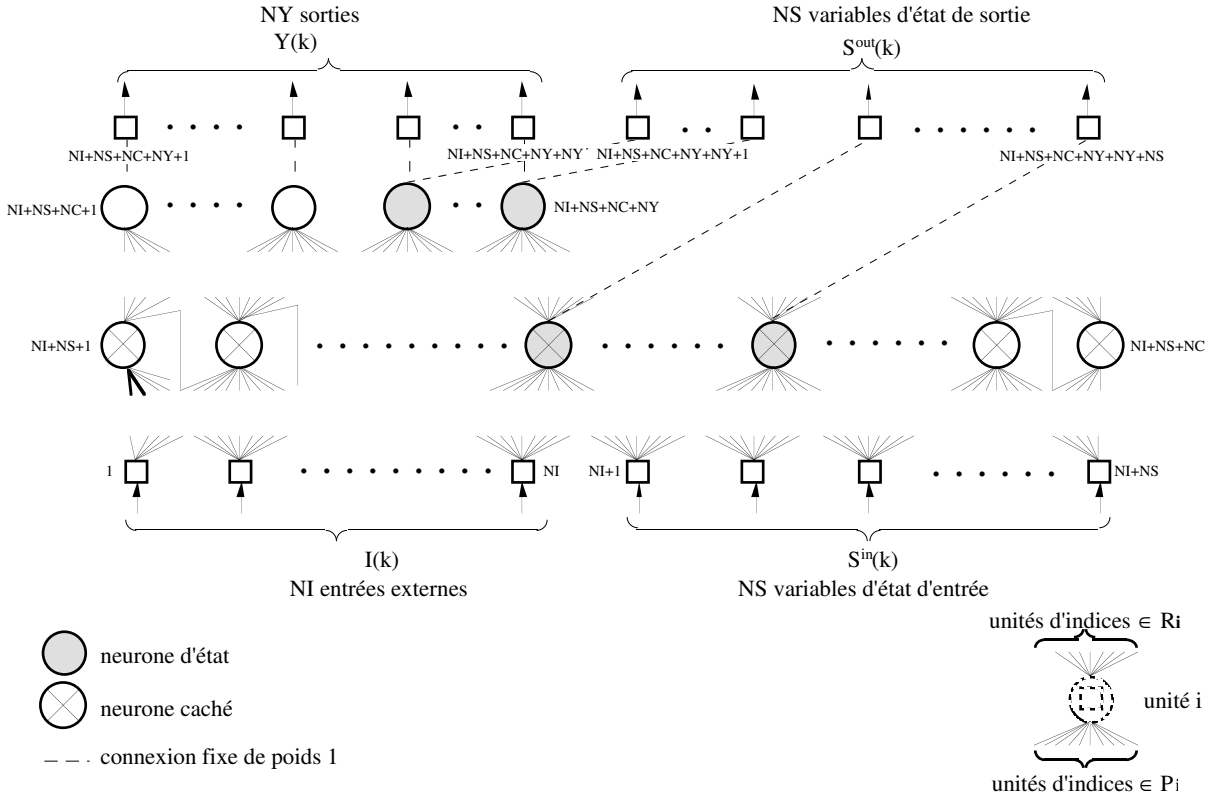

Figure 2.
Détail de la copie n°k utilisée pour l'apprentissage.

Pour faciliter la présentation modulaire du calcul du gradient, nous distinguons les neurones d'unités particulières dont la fonction d'activation est l'identité, appelés cellules (représentées par des carrés sur la figure 2). Les entrées sont ainsi des cellules d'entrée. Les valeurs des sorties de la copie sont les valeurs des activités de NY cellules de sortie, connectées aux NY neurones de sortie avec un coefficient 1. De même, les valeurs des variables d'état de sortie sont les valeurs des activités de NS cellules d'état de sortie, connectées aux NS neurones d'état avec un coefficient 1 (un neurone d'état

est un neurone caché ou un neurone de sortie). Chaque copie est ainsi composée de $NI+NS+N_C+NY+NY+NS$ unités :

- NI cellules d'entrées externes (numérotées de 1 à NI) ;
- NS cellules d'état d'entrée (numérotées de NI+1 à NI+NS) ;
- NC neurones cachés (ordonnés de NI+NS+1 à NI+NS+NC) ;
- NY neurones de sortie (ordonnés de NI+NS+NC+1 à NI+NS+NC+NY) ;
- NY cellules de sortie (numérotées de NI+NS+NC+NY+1 à NI+NS+NC+NY+NY) ;
- NS cellules d'état de sortie (numérotées de NI+NS+NC+NY+NY+1 à NI+NS+NC+NY+NY+NS).

Il est commode de définir la connectivité du réseau par les P_i, ensemble des indices des unités *p*ropageant leur activité vers l'unité i, et les R_i, ensemble des unités *r*ecevant l'activité du neurone i. En particulier, l'ensemble P_i d'une cellule d'état de sortie est constitué de l'indice du neurone d'état auquel elle correspond, et l'ensemble P_i d'une cellule de sortie est constitué de l'indice du neurone de sortie auquel elle correspond. De même, l'ensemble R_i d'un neurone de sortie est constitué de l'indice de la cellule de sortie à laquelle il correspond, et, si c'est aussi un neurone d'état, de l'indice de la cellule d'état de sortie à laquelle il correspond.

Exemple.

À titre illustratif, la forme canonique associée à un prédicteur bouclé entrée-sortie de la forme :

$$y(k+1) = \varphi_{RN}\left(y(k), \ldots, y(k-n+1), u(k), \ldots, u(k-m+1); C\right)$$

est représentée sur la figure 3, avec les mêmes notations que celles de la figure 2.

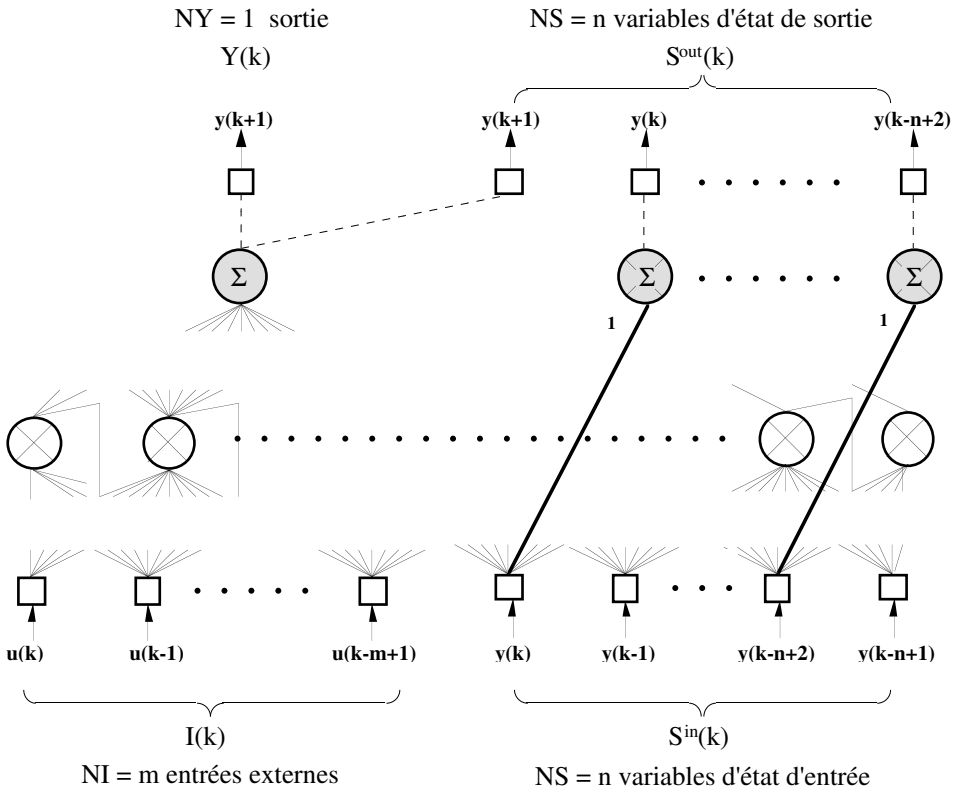

Figure 3.
Copie n°k utilisée pour l'apprentissage d'un prédicteur bouclé entrée-sortie.

Le neurone de sortie est linéaire, ainsi que les neurones d'état. Ils ne possèdent qu'une connexion de poids 1 fixé sur les variables d'état d'entrée (valeurs passées de la sortie) auxquelles ils correspondent. Dans la suite, nous omettons ces neurones sur les schémas (voir les notations du chapitre 2 §I.1, le chapitre 3, et les notations du chapitre 5).

I. CALCUL DE LA FONCTION DE COÛT ET DE SON GRADIENT.

Calcul de la fonction de coût par propagation.

La propagation consiste à calculer les valeurs des potentiels $\{v_i(k)\}$ et des activités $\{z_i(k)\}$ de toutes les unités du réseau ainsi que les erreurs de sortie $\{E_i(k)\}$ à chaque instant k, donc pour chaque copie (k est le numéro de la copie), de l'instant 1 à l'instant N. On peut ainsi calculer la fonction de coût définie pour une itération, dont nous rappelons l'expression en omettant l'indice de l'itération :

$$J(C) = \frac{1}{2} \sum_{k=1}^{N} \sum_{i=1}^{NY} w_{ii}\, E_i^2(k)$$

Calcul du gradient de la fonction de coût par rétro-propagation.

La rétro-propagation consiste à calculer à chaque itération de l'algorithme les valeurs $\{\partial J/\partial v_i(k)\}$ des dérivées partielles de la fonction de coût par rapport au potentiel de chaque unité du réseau à tout instant k, de la dernière unité de chaque copie à la première, de l'instant N à l'instant 1, d'où la dénomination de rétro-propagation[1]. Ces dérivées permettent ensuite le calcul des dérivées de la fonction de coût par rapport aux coefficients. En effet, écrivons la différentielle de J(C) :

$$dJ(C) = \sum_{i,\,j} \sum_{k=1}^{N} \frac{\partial J}{\partial c_{ij}(k)}\; dc_{ij}(k) \quad \text{avec } dc_{ij}(k) = dc_{ij}\ \forall k, \text{ soit } \quad dJ(C) = \sum_{i,\,j} \sum_{k=1}^{N} \frac{\partial J}{\partial c_{ij}(k)}\; dc_{ij}$$

où $c_{ij}(k)$ représente le coefficient c_{ij} dans la copie numéro k. $c_{ij}(k)$ n'est qu'une écriture, puisque les coefficients sont les mêmes pour chaque copie de la fenêtre de la fonction de coût, donc pour chaque instant k. La composante du gradient de J relative au coefficient c_{ij} s'écrit :

$$\left(\frac{\partial J}{\partial c_{ij}}\right)_{\substack{\forall p,q\neq i,j \\ c_{pq}\,constant}} = \sum_{k=1}^{N} \left(\frac{\partial J}{\partial c_{ij}(k)}\right)_{\substack{\forall p,q\neq i,j \\ c_{pq}\,constant}}$$

La rétro-propagation consiste à poser le calcul de la manière suivante :

$$\frac{\partial J}{\partial c_{ij}(k)} = \frac{\partial J}{\partial v_i(k)} \frac{\partial v_i(k)}{\partial c_{ij}(k)} = \frac{\partial J}{\partial v_i(k)}\, z_j(k)$$

---

[1] Le calcul par rétro-propagation suppose implicitement la nullité des dérivées de la fonction de coût par rapport aux entrées d'état de la première copie. Pour leur imposer des valeurs non nulles, il faut calculer le gradient par une méthode directe [NER92a]. Or imposer des valeurs non nulles n'est justifié que pour un apprentissage récursif, ce qui ne sera pas le cas dans ce mémoire, où nous traitons l'apprentissage de systèmes non adaptatifs. Nous utilisons donc exclusivement le calcul par rétropropagation, beaucoup plus économe que le calcul direct en nombre d'opérations.

où P_i est l'ensemble des indices des unités propageant leur activité au neurone i. Les valeurs des activités $\{z_j(k)\}$ ont toutes été calculées lors de la propagation. La connaissance des $\{\partial J/\partial v_i(k)\}$ et des $\{z_j(k)\}$ permet donc le calcul du gradient.

Si le réseau n'est pas bouclé (NS=0), chaque vecteur d'erreur E(k) est calculé indépendamment des autres puisque les copies ne se transmettent aucune valeur. De même, la contribution de chaque erreur au gradient est calculée de façon indépendante. La contribution de la copie k à la valeur du gradient dépend des entrées externes I(k) et des valeurs désirées D(k) (N rétro-propagations).

Si le réseau est bouclé, les valeurs des variables d'état sont transmises d'une copie à la suivante :

$$S_i^{in}(k) = S_i^{out}(k\text{-}1)$$

La contribution de chaque erreur au gradient ne peut plus être calculée de façon indépendante. En effet, on a :

$$\frac{\partial J}{\partial S_i^{out}(k)} = \frac{\partial J}{\partial S_i^{in}(k+1)}$$

On effectue dans ce cas une seule rétro-propagation.

I.1. CALCUL DE LA FONCTION DE COÛT, OU PROPAGATION.

* Valeurs des entrées externes : *par définition*, pour i=1 à NI

$$z_i(k) = v_i(k) = I_i(k)$$

* Valeurs des activités des cellules d'état d'entrée : *par définition*, pour i=1 à NS

$$k = 1 : \quad z_{NI+i}(1) = v_{NI+i}(1) = S_i^{in}(1) = \text{valeur imposée}$$

$$k > 1 : \quad z_{NI+i}(k) = v_{NI+i}(k) = S_i^{in}(k) = S_i^{out}(k\text{-}1)$$

La valeur imposée est choisie en fonction du problème particulier considéré, cf chapitres 3 et 5.

* Valeurs des activités des neurones cachés, des neurones de sortie, des cellules de sortie, et des cellules d'état de sortie : *calcul*, pour i variant de NI+NS+1 à NI+NS+NC+NY+NY+NS

$$z_i(k) = f_i\left(v_i(k)\right) \text{ où } \quad v_i(k) = \sum_{j \in P_i} c_{ij}\, z_j$$

où P_i est l'ensemble des indices des unités propageant leur activité au neurone i.

* Valeurs des sorties : *par définition,* pour i=1 à NY

$$Y_i(k) = z_{NI+NS+NC+NY+i}(k)$$

* Valeurs des variables d'état de sortie : *par définition*, pour i=1 à NS

$$S_i^{out}(k) = z_{NI+NS+NC+NY+NY+i}(k)$$

* Valeurs des erreurs : *calcul*, pour i=1 à NY de

$$E_i(k) = D_i(k) - Y_i(k)$$

I.2. CALCUL DU GRADIENT DE LA FONCTION DE COÛT, OU RÉTRO-PROPAGATION.

* Valeurs des $\partial J/\partial v_i(k)$ des cellules d'état de sortie : *par définition*, pour i=1 à NS

$$k = N : \quad \frac{\partial J}{\partial v_{NI+NS+NC+NY+NY+i}(N)} = \frac{\partial J}{\partial S_i^{out}(N)} = 0$$

$$k < N : \quad \frac{\partial J}{\partial v_{NI+NS+NC+NY+NY+i}(k)} = \frac{\partial J}{\partial S_i^{out}(k)} = \frac{\partial J}{\partial S_i^{in}(k+1)}$$

* Valeurs des $\partial J/\partial v_i(k)$ des cellules de sortie : *calcul*, pour i=1 à NY

$$\frac{\partial J}{\partial v_{NI+NS+NC+NY+i}(k)} = - w_{ii} E_i(k)$$

* Valeurs des $\partial J/\partial v_i(k)$ des neurones de sortie, des neurones cachés et des cellules d'état d'entrée : *calcul*, pour i décroissant de NI+NS+NC+NY à NI+1

$$\frac{\partial J}{\partial v_i(k)} = f_i^{'}\left(v_i(k)\right) \sum_{h \in R_i} c_{hi} \frac{\partial J}{\partial v_h(k)}$$

où R_i est l'ensemble des indices des unités recevant l'activité du neurone i.

* Valeur de $\partial J/\partial S^{in}$: *par définition*, pour i=1 à NS

$$\frac{\partial J}{\partial S_i^{in}(k)} = \frac{\partial J}{\partial v_{NI+i}(k)}$$

L'algorithme de calcul de la fonction de coût et de son gradient est dit " dirigé " si le réseau est non bouclé (NS=0), " semi-dirigé " sinon. Ceci signifie, dans le premier cas, que le réseau n'a pas d'état, et qu'il est donc entièrement " dirigé " par les entrées externes, et, dans le second cas, que l'état du réseau ne lui est imposé qu'au début de la fenêtre de la fonction d'apprentissage (pour la première copie)[2]. Les notions de dirigé et de semi-dirigé prendront tout leur sens dans les chapitres consacrés à l'apprentissage de prédicteurs et de correcteurs.

I.3. RÉSUMÉ DES CALCULS.

La figure 4 résume les calculs à effectuer pour chaque copie k, k variant de 1 à N, horizon sur lequel la fonction de coût est définie. La propagation, ou calcul de la fonction de coût par le réseau de propagation FF, doit être effectuée dans tous les blocs avant la rétro-propagation, ou calcul du gradient de la fonction de coût par le réseau BP, sauf si le réseau n'est pas bouclé (les flèches grisées disparaissent).

---

[2] Dans le cas récursif, un troisième algorithme, dit " non-dirigé ", est également utilisé [NER92a].

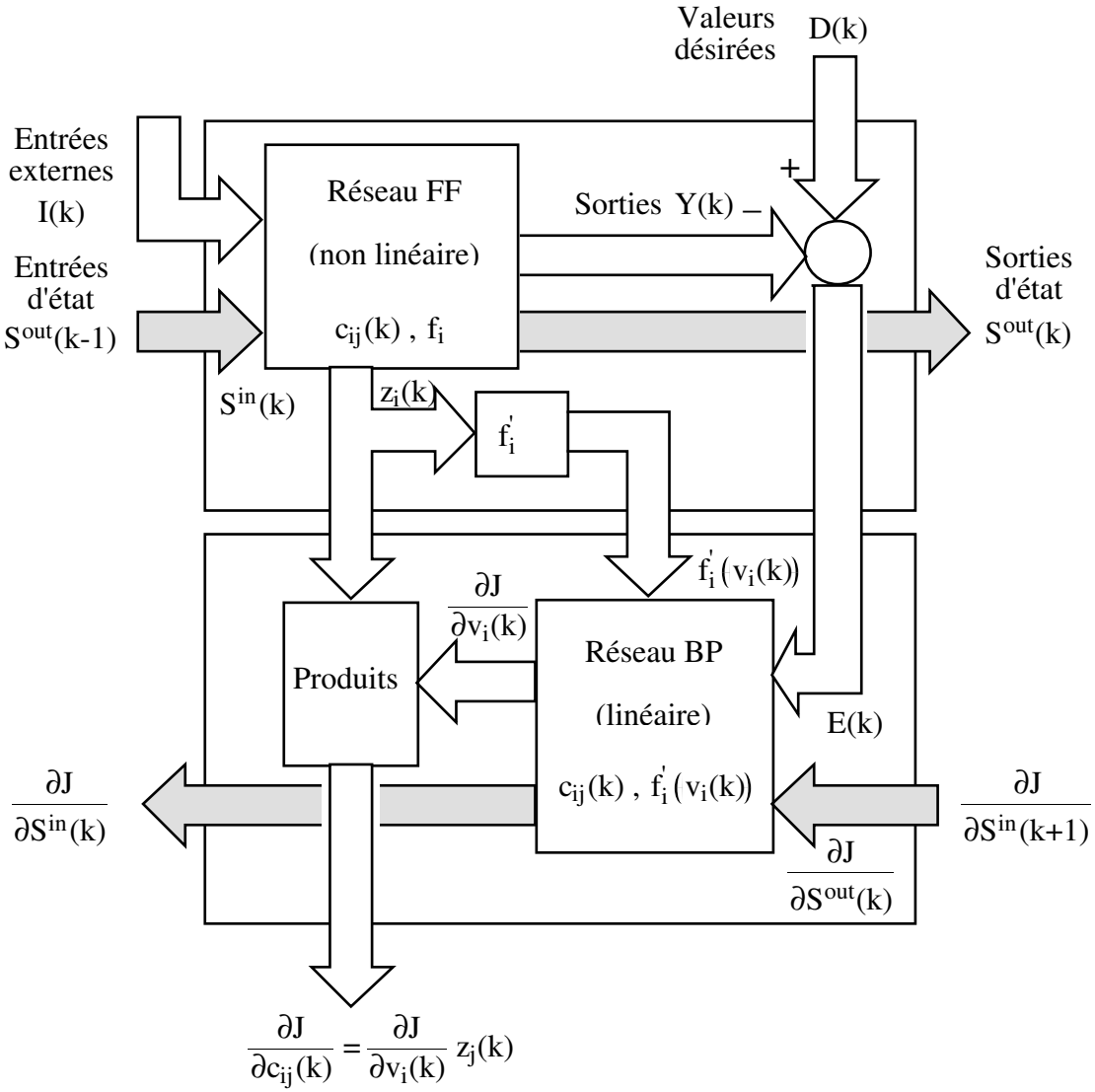

Figure 4.
Calculs correspondants à la copie k.

II. MODIFICATION DES COEFFICIENTS.

Dans le cas général, le calcul de la modification des coefficients (dont la valeur n'est pas fixée[3]) à l'itération i s'écrit :

$$\Delta C = + \mu D$$

où D est une direction de descente,
$\mu>0$ est le pas de descente.

[3] Dans le cas de l'apprentissage d'un correcteur en partculier, le système d'apprentissage utilise un modèle de simulation, dont les coefficients ne doivent pas être modifiés.

II.1. MÉTHODE DE GRADIENT À PAS CONSTANT.

La méthode la plus couramment utilisée est celle où D est l'opposé du gradient et μ est constant. Les modifications des coefficients sont données par l'expression :

$$\Delta c_{ij} = - \mu \frac{\partial J}{\partial c_{ij}} = - \mu \sum_{k=1}^{N} \frac{\partial J}{\partial c_{ij}}(k) = - \mu \sum_{k=1}^{N} \frac{\partial J}{\partial v_i(k)} z_j(k)$$

Cette méthode présente l'inconvénient d'avoir une vitesse de convergence très lente lorsque l'on s'approche du minimum. Nous utilisons les méthodes suivantes qui permettent d'accélérer la convergence de façon appréciable.

II.2. MÉTHODES À PAS VARIABLE.

Quelle que soit la direction de descente utilisée, il est possible d'asservir le pas μ de telle sorte que la fonction de coût diminue à chaque modification des coefficients. Nous avons utilisé deux méthodes de minimisation unidimensionnelle " économiques ", les méthodes de Nash [NAS90] et de Wolfe et Powell [MIN83]. Ces méthodes permettent d'obtenir un pas convenable avec un nombre très limité d'évaluations de la fonction de coût et du gradient J. Notons C_i les coefficients, D_i la direction de descente, et μ_i le pas de descente à l'itération i. Soit la fonction g définie par : $g\ (\mu_i) = J\left(C_i + \mu_i D_i\right)$. Sa dérivée s'écrit : $g'(\mu_i) = D_i^T \nabla J\left(C_i + \mu_i D_i\right)$. g est donc la fonction dont on cherche le minimum en fonction du pas μ_i.

a) Règle de Nash.

Le principe de cette méthode est que le pas de descente ne doit pas être choisi trop grand sinon l'algorithme risque d'avoir un comportement oscillatoire. Le pas de descente μ_i doit être choisi de façon telle que :

$$g\left(\mu_i\right) \le g\left(0\right) + m_1\, \mu_i\, g'\left(0\right)$$

soit :

$$J\left(C_i + \mu_i D_i\right) \le J\left(C_i\right) + m_1\, \mu_i\, D_i^T \nabla J\left(C_i\right)$$

Cette condition assure la propriété de descente ; m_1, le facteur de tolérance, est un nombre petit devant 1 (par exemple 10^{-3}). Si la condition ci-dessus n'est pas satisfaite, le pas μ_i est multiplié par un facteur de réduction (Nash propose 0,2). En pratique, avec les valeurs précédentes, on se limite à une vingtaine d'évaluations de J, après quoi les modifications effectuées sont de l'ordre de grandeur des erreurs d'arrondis.

b) Règle de Wolfe et Powell.

(a) Le pas de descente ne doit pas être choisi trop grand sinon l'algorithme risque d'avoir un comportement oscillatoire.

(b) Le pas de descente ne doit pas être choisi trop petit pour permettre une convergence rapide de l'algorithme.

μ_i doit être choisi de façon telle que :

(1) $g(\mu_i) \leq g(0) + m_1 \mu_i g'(0)$ avec $m_1 \in]0,1[$

Cette première règle assure la propriété de descente.

(2) $g'(\mu_i) \geq m_2 g'(0)$ avec $m_2 \in]m_1,1[$

La seconde règle permet de s'assurer que le pas n'est pas non plus choisi trop petit, comme le montre la figure 5. Les deux règles s'écrivent, en fonction du critère et de son gradient :

(1) $J(C_i + \mu_i D_i) \leq J(C_i) + m_1 \mu_i D_i^T \nabla J(C_i)$ avec $m_1 \in]0,1[$

(2) $D_i^T \nabla J(C_i + \mu_i D_i) \geq m_2 D_i^T \nabla J(C_i)$ avec $m_2 \in]m_1,1[$

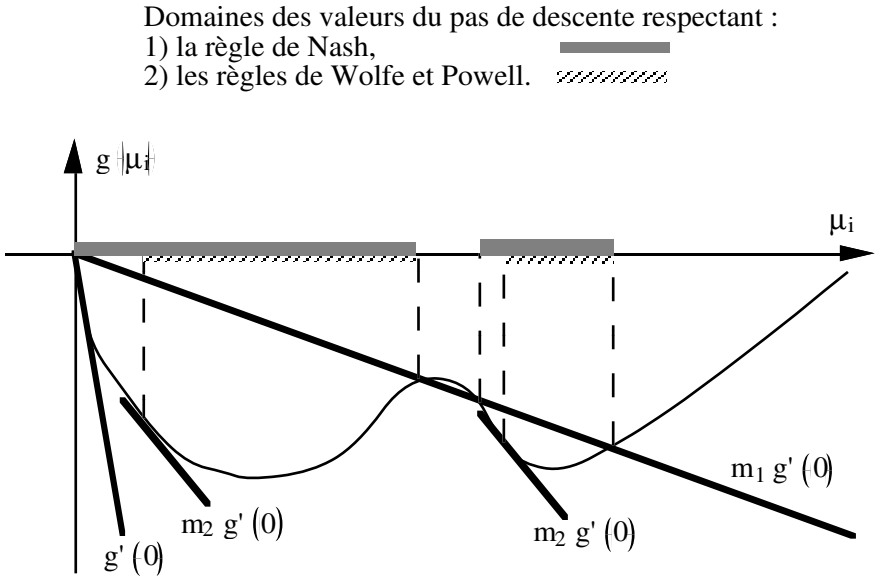

Figure 5.
Optimisation unidimensionnelle : règles de Nash et de de Wolfe et Powell (coupe dans la direction de descente D_i).

Pratiquement, la procédure est la suivante :

Initialiser les valeurs : $\mu=1$; $\mu_{min}=\mu^0_{min}$, ; $\mu_{max}=\mu^0_{max}$.

a) Calculer $J(C_i + \mu D_i)$.

b) Si la condition (1) est vérifiée, aller en c); sinon $\mu_{max}=\mu$ (μ est trop grand) et aller en d).

c) Si la condition (2) est vérifiée $\mu_i=\mu$ est accepté ; sinon $\mu_{min}=\mu$ (μ est trop petit) et aller en d).

d) Modifier le pas μ : $\mu = \frac{\mu_{min} + \mu_{max}}{2}$ et aller en a).

On prend par exemple : $m_1 = 0,1$; $m_2 = 0,7$; $\mu_{min}=0$; $\mu_{max}=10$ (ce choix n'est pas très critique).

La mise en œuvre de cette méthode est donc un peu plus lourde que la précédente.

II.3. MÉTHODE QUASI NEWTONIENNE À PAS VARIABLE.

Nous avons également employé une méthode utilisant une autre direction de descente que le gradient, une méthode quasi newtonienne (ou méthode à métrique variable) utilisant l'algorithme de Broyden, Fletcher, Goldfarb et Shanno (BFGS) dont la vitesse de convergence est beaucoup plus grande que celle du gradient [MIN83].

Méthode de Newton.

La méthode de Newton consiste à remplacer la fonction de coût par son approximation quadratique au voisinage du point courant (la quadrique osculatrice) :

$$q(C) = J(C_i) + \nabla J^T(C_i)(C - C_i) + \frac{1}{2}(C - C_i)^T \nabla^2 J(C_i)(C - C_i)$$

et à choisir les coefficients C_{i+1} de manière à minimiser q(C). Le minimum existe si le Hessien $\nabla^2 J(C_i)$ est défini positif. Ce qui conduit à la formule itérative :

$$C_{i+1} = C_i - \left[\nabla^2 J(C_i)\right]^{-1} \nabla J(C_i)$$

La méthode de Newton demande donc de calculer puis d'inverser le Hessien à chaque itération. Notons que le pas de descente est fixé (μ=1). Appliquée à une fonction J des coefficients C quadratique strictement convexe, cette méthode converge en une seule itération, mais dans le cas général d'une fonction non linéaire quelconque, elle ne possède pas la propriété de convergence globale. Si les coefficients de départ sont trop éloignés du minimum, la méthode peut ne pas converger.

Méthode de quasi-Newton.

Le principe des méthodes quasi newtoniennes consiste en une généralisation de la formule itérative de Newton :

$$C_{i+1} = C_i + \mu_i D_i \text{ avec } D_i = - H_i \nabla J(C_i)$$

où H_i est une matrice définie positive donnant la direction de descente à partir du gradient $\nabla J(C_i)$, et μ_i est choisi tel que $J(C_i + \mu_i D_i) < J(C_i)$, donc par exemple par les méthodes de Nash ou de Wolfe et Powell. La matrice H_i est une approximation de l'inverse du Hessien. En effet, elle est modifiée à chaque itération de telle manière que, pour une fonction quadratique, elle converge vers l'inverse du Hessien $\left[\nabla^2 J(C_i)\right]^{-1}$. Différentes formules du type :

$$H_{i+1} = H_i + \Delta_i$$

ont été proposées. L'une d'elle est la formule BFGS suivante :

$$H_{i+1} = H_i + \left[1 + \left(\frac{\gamma_i^T H_i \gamma_i}{\Delta C_i^T \gamma_i}\right)\right] \frac{\Delta C_i^T \Delta C_i}{\Delta C_i^T \gamma_i} - \frac{\Delta C_i \gamma_i^T H_i + H_i \gamma_i \Delta C_i^T}{\Delta C_i^T \gamma_i}$$

avec $\gamma_i = \nabla J(C_{i+1}) - \nabla J(C_i)$ et $\Delta C_i = C_{i+1} - C_i$.

Si la matrice H_{i+1} ainsi calculée n'est pas définie positive, elle est réinitialisée à la matrice identité. On repart alors avec $D_i = - \nabla J(C_i)$, c'est-à-dire dans la direction opposée à celle du gradient. Un des principaux attraits de la méthode BFGS est qu'elle est relativement insensible au choix du pas. Comme nous l'avons précisé ci-dessus, il suffit que μ_i soit tel que $J(C_i + \mu_i D_i) < J(C_i)$. On peut donc se contenter de la méthode de Nash, la plus simple des méthodes présentées.

Annexe II

SYSTÈMES DE COMMANDE LINÉAIRE PAR SIMPLE BOUCLAGE ET AVEC MODÈLE INTERNE

INTRODUCTION.

Cette annexe établit l'expression des correcteurs-S et -D dans le cas de modèles discrets linéaires mono-entrée/mono-sortie du processus. Puis elle étudie les propriétés de systèmes de commande utilisant ces correcteurs, des systèmes de commande par simple bouclage (SCSB), ou avec modèle interne (SCMI). Le but de cette étude est, outre une récapitulation utile des résultats en linéaire, leur généralisation partielle au cas de modèles et de correcteurs non linéaires, par exemple des réseaux de neurones, généralisation mise à profit au chapitre 5.

I. CORRECTEURS-S ET -D.

Les correcteurs-S et -D sont particuliers aux modèles à temps discret. Soit un système de commande constitué d'un tel modèle de retard d, en simple bouclage avec le correcteur-S ou -D. Les définitions des correcteurs sont les suivantes :

Correcteur-S : il impose au système de commande une sortie de référence.

Soit une trajectoire de référence $\{y_r(k)\}$. Le correcteur-S délivre à chaque instant k une commande telle que, quel que soit l'état du modèle à cet instant, et si aucune perturbation n'intervient ensuite, *la sortie du système de commande est égale à la sortie de référence* à partir de l'instant k+d :

$$y(k') = y_r(k') \qquad \forall k' \geq k+d$$

Correcteur-D : il impose au système de commande une dynamique de référence.

Soit une trajectoire de consigne {r(k)}, et une dynamique donnée par un modèle de référence linéaire de retard d de sortie y_r telle que $E(q)\, y_r(k+d) = H(q)\, r(k)$, ou E et H sont deux polynômes en q^{-1}. Le correcteur-D délivre à chaque instant k une commande telle que, quel que soit l'état du modèle à cet instant, et si aucune perturbation n'intervient ensuite, *la dynamique du système de commande est égale à la dynamique de référence* à partir de l'instant k+d :

$$E(q)\, y(k') = q^{-d}\, H(q)\, r(k') \qquad \forall k' \geq k+d$$

Notons que le correcteur-S est un cas particulier du correcteur-D, avec $E(q) = H(q) = 1$. Cependant, sa mise en œuvre est suffisamment particulière pour que nous traitions son cas séparément du cas du correcteur-D : en effet, elle nécessite un modèle de référence au sein des systèmes de commande qui l'utilisent pour calculer la trajectoire de référence $\{y_r(k)\}$ à partir de la consigne (voir chapitre 5).

Dans le cas linéaire, il n'est pas nécessaire de traiter le cas des modèles d'état, que l'on peut toujours écrire sous une forme entrée-sortie adéquate [GOO84]. Mais le le but de cette présentation est une généralisation au cas non linéaire : or le modèle non linéaire disponible peut être du type représentation d'état. La mise sous forme entrée-sortie de ce modèle demanderait l'apprentissage d'un modèle entrée-sortie, apprentissage dont les difficultés ont été soulevées au chapitre 2 §I.2.2.3 (ordre du modèle entrée-sortie…). Nous étudions donc aussi l'expression des correcteurs dans le cas où l'on dispose d'un modèle d'état. Les expressions des correcteurs dans le cas d'un modèle linéaire entrée-sortie sont données dans [GOO84].

I.1. EXPRESSIONS DES CORRECTEURS DANS LE CAS D'UN MODÈLE ENTRÉE-SORTIE.

Nous considérons le modèle discret linéaire entrée-sortie de retard d :

$$A(q)\, y(k) = B(q)\, u(k)$$

avec :

$A(q) = 1 + a_1\, q^{-1} + \ldots + a_n\, q^{-n}$

$B(q) = q^{-d}\, B'(q) = q^{-d}\left(b_0 + b_1\, q^{-1} + \ldots + b_{m'}\, q^{-m'}\right)$

où d est le retard du modèle. Le degré du polynôme B est $d + m' = m$.

I.1.1. Correcteur-S.

Comme le modèle possède un retard d, la commande u(k) affecte la sortie à l'instant k+d. Cherchons les arguments en fonction desquels s'exprime la sortie y(k+d) :

- pour calculer y(k+1), on a besoin de y(k), … , y(k-n+1), u(k-d+1), … , u(k-d-m'+1);
- pour calculer y(k+2), on a besoin, en plus, de u(k-d+2);

…

- pour calculer y(k+d), on a besoin, en plus, de u(k), d'où:

$$\begin{aligned} y(k+d) &= \varphi\left(y(k), \ldots, y(k-n+1), u(k), \ldots, u(k-d-m'+1)\right) \\ &= \varphi\left(y(k), \ldots, y(k-n+1), u(k), \ldots, u(k-m+1)\right) \end{aligned}$$

En linéaire, il est facile de calculer la fonction φ, ou prédicteur à d pas de la sortie. Le raisonnement précédent montre qu'il faut effectuer la division 1/A limitée aux d premiers termes :

$\frac{1}{A(q)} = F(q) + q^{-d}\frac{G(q)}{A(q)}$ soit : $1 = F(q)A(q) + q^{-d}\, G(q)$

où F, monique, et G sont les uniques polynômes de degrés d-1 et n-1 satisfaisant cette égalité :

$F(q) = 1 + f_1\, q^{-1} + \ldots + f_{d-1}\, q^{-d+1}$
$G(q) = g_0 + g_1\, q^{-1} + \ldots + g_{n-1}\, q^{-n+1}$
Les coefficients de F et de G sont :
$f_0 = 1$; $f_i = -\sum_{j=0}^{i-1} f_j\, a_{i-j}$ pour i=1 à d-1 ; $g_i = -\sum_{j=0}^{d-1} f_j\, a_{i+d-j}$ pour i=0 à n-1 .

Pour obtenir l'expression du prédicteur, on multiplie celle du modèle par le polynôme F :
$F(q)\, A(q)\, y(k) = F(q)\, q^{-d}\, B'(q)\, u(k)$; comme $F(q)A(q) = 1 - q^{-d}\, G(q)$:
$y(k) = q^{-d}\, G(q)\, y(k) + q^{-d}\, F(q)\, B'(q)\, u(k)$ soit :

$$y(k+d) = G(q)\, y(k) + F(q)\, B'(q)\, u(k)$$

qui est l'expression du prédicteur cherché, encore appelé prédicteur à d pas.

Le correcteur-S est le correcteur qui permet d'obtenir $y(k+d)=y_r(k+d)$; il s'écrit donc :

$$\boxed{u(k) = y_r(k+d) - G(q)\, y(k) + \left(1 - F(q)\, B'(q)\right) u(k)}$$

où G est de degré n-1 et 1–FB' de degré $d-1+m' = m-1$. Ce correcteur est en général bouclé (dès que m>1). Si cette commande est appliquée à l'instant k, et si aucune perturbation n'intervient ensuite, le système bouclé obéit à :

$$B(q)\, u(k) = A(q)\, y_r(k) \qquad \forall k' \geq k+d$$

Le correcteur-S est donc stable si les racines du polynôme B sont à l'intérieur du cercle unité, c'est-à-dire si le modèle est *à minimum de phase*.

Remarque.
Tous les polynômes utilisés dans les expressions de prédicteurs et de correcteurs ont un terme constant (en q^0).

I.1.2. Correcteur-D.

Soit la dymamique de référence donnée par le modèle :

$$E(q)\, y_r(k+d) = H(q)\, r(k)$$

avec :
$E(q) = 1 + e_1\, q^{-1} + \ldots + e_p\, q^{-p}$
$H(q) = h_0 + h_1\, q^{-1} + \ldots + h_p\, q^{-p}$
Le prédicteur nécessaire à la synthèse du correcteur-D s'obtient en effectuant la division de E par A limitée aux d premiers termes :

$$\frac{E(q)}{A(q)} = F(q) + q^{-d}\frac{G(q)}{A(q)} \quad soit \quad E(q) = F(q)\, A(q) + q^{-d}\, G(q)$$

où F, monique, et G sont les deux seuls polynômes de degrés d-1 et n-1 satisfaisant l'équation ci-dessus (ceci suppose p≤d+n-1). Les coefficients de F et de G sont :

$f_0 = 1$; $f_i = -\sum_{j=0}^{i-1} f_j\, a_{i-j} - e_i$ pour i=1 à d-1 ; $g_i = -\sum_{j=0}^{d-1} f_j\, a_{i+d-j} - e_{i+d}$ pour i=0 à n-1 .

Pour obtenir l'expression du prédicteur associé au problème, on multiplie comme précédemment celle du modèle par le polynôme F :
$F(q)\, A(q)\, y(k) = F(q)\, q^{-d}\, B'(q)\, u(k)$; comme $F(q)\, A(q) = E(q) - q^{-d}\, G(q)$:

$E(q)\ y(k) = q^{-d}\ G(q)\ y(k) + q^{-d}\ F(q)\ B'(q)\ u(k)$ soit :

$$E(q)\ y(k+d) = G(q)\ y(k) + F(q)\ B'(q)\ u(k)$$

qui est l'expression du prédicteur cherché.

Le correcteur-D est le correcteur qui permet d'obtenir $E(q)\ y(k+d) = H(q)\ r(k)$: il s'écrit donc :

$$\boxed{u(k) = H(q)\ r(k) - G(q)\ y(k) + \left(1 - F(q)\ B'(q)\right) u(k)}$$

G est un polynôme en q^{-1} de degré n-1et 1–FB' est de degré m–1. Le correcteur est en général bouclé (dès que m>1). Si cette commande est appliquée à l'instant k, et si aucune perturbation n'intervient ensuite, le système bouclé obéit à :

$$E(q)\ B(q)\ u(k) = E(q)\ A(q)\ y_r(k) \qquad \forall k' \geq k+d$$

Pour que le correcteur-D soit stable, il faut donc aussi que le modèle soit *à minimum de phase*.

I.2. Expressions des correcteurs dans le cas d'un modèle d'état.

Comme nous l'avons précisé dans l'introduction, nous traitons le problème dans le cas d'une représentation d'état uniquement en vue de la généralisation aux modèles non linéaires. Nous considérons le modèle discret linéaire d'état suivant :

$$\begin{cases} x(k+1) = A\ x(k) + B\ u(k) \\ y(k) = C\ x(k) \end{cases}$$

où A est une matrice n x n, B une matrice colonne n x 1, et C une matrice ligne 1 x n. Son ordre relatif est d.

I.2.1. Correcteur-S.

Comme dans le cas entrée-sortie, cherchons le prédicteur de la sortie du modèle à d pas, ou d est l'ordre relatif du modèle. Pour un modèle mono-entrée/mono-sortie linéaire ou non, l'ordre relatif est défini comme le retard existant entre l'entrée et la sortie du modèle, c'est-à-dire l'entier d tel que, pour x(k) et u(k) donnés, l'effet de l'entrée u(k) apparaît à la sortie après d pas :

$y(k+1) = CA\ x(k) + CB\ u(k)$

$y(k+2) = CA^2\ x(k) + CAB\ u(k) + CB\ u(k+1)$

…

$$y(k+i) = CA^i\ x(k) + \sum_{j=1}^{i} CA^{j-1}B\ u(k+i-j)$$

L'ordre relatif d est donc la plus petite valeur entière de j telle que : $CA^{j-1}B \neq 0$. Pour un modèle mono-entrée/mono-sortie, $CA^{d-1}B$ est un scalaire. On a :

$$y(k+d) = CA^d\ x(k) + CA^{d-1}B\ u(k)$$

qui est l'expression du prédicteur à d pas pour un modèle d'état.

Le correcteur-S est le correcteur qui permet d'obtenir $y(k+d)=y_r(k+d)$. Son expression est donc :

$$\boxed{u(k) = \frac{1}{CA^{d-1}B}\left(y_r(k+d) - CA^d\ x(k)\right)}$$

Ce correcteur n'est pas bouclé, sa sortie n'est fonction que du signal de référence et de l'état du modèle. La dynamique du système {correcteur+modèle} est :

$$x(k+1) = \left(A - \frac{BCA^{d}}{CA^{d-1}B}\right) x(k) + \frac{B}{CA^{d-1}B}\, y_r(k+d)$$

Pour que la commande soit applicable (bornée), il est nécessaire que la matrice :

$$A - \frac{BCA^{d}}{CA^{d-1}B}$$

ait des valeurs propres de module inférieur à 1. Cela signifie en particulier que le modèle doit être *à minimum de phase*.

I.2.2. Correcteur-D.

Par définition du correcteur-D, on désire obtenir :

$$E(q)\, y(k+d) = H(q)\, r(k)$$

soit :

$y(k+d) + e_1\, y(k+d-1) + \ldots + e_p\, y(k+d-p) = H(q)\, r(k)$

Calculons chacun des y en fonction de l'état à l'instant k, quand c'est possible. On a :

$y(k) = C\, x(k)$

$y(k+1) = CA\, x(k)$

…

$y(k+d) = CA^{d}\, x(k) + CA^{d-1}B\, u(k)$

Il faut donc que :

$$CA^{d}\, x(k) + CA^{d-1}B\, u(k) + \ldots + e_d\, C\, x(k) + e_{d+1}\, y(k-1) + \ldots + e_p\, y(k+d-p) = H(q)\, r(k)$$

L'expression du correcteur est :

- si p > d :

$$u(k) = \frac{1}{CA^{d-1}B}\left[H(q)\, r(k) - \left(CA^{d} + e_1\, CA^{d-1} + \ldots + e_d\, C\right) x(k) - \left(e_{d+1}\, y(k-1) + \ldots + e_p\, y(k+d-p)\right)\right]$$

- si p ≤ d :

$$u(k) = \frac{1}{CA^{d-1}B}\left[H(q)\, r(k) - \left(CA^{d} + e_1\, CA^{d-1} + \ldots + e_p\, C\, A^{d-p}\right) x(k)\right]$$

Ce correcteur n'est pas bouclé : ses arguments sont des valeurs successives du signal de référence (pondérées par le polynôme H), l'état du modèle, et un certain nombre de valeurs de ses sorties antérieures. Si p > d, il est impossible d'exprimer u(k) uniquement en fonction de la consigne et de l'état[4].

---

[4] Pour une expression du correcteur-D formulée comme un retour d'état statique, voir [GOO84], p. 148 et suivantes : l'état choisi s'exprime comme une combinaison linéaire des sorties et des commandes passées. Nous préférons la solution proposée ci-dessus, qui fournit dans tous les cas un correcteur non bouclé, et donc plus facile à réaliser par apprentissage avec un réseau de neurones (on peut utiliser un algorithme dirigé).

II. SYSTÈMES DE COMMANDE POUR LA POURSUITE ET LA RÉGULATION.

Nous étudions les propriétés des systèmes de commande par simple bouclage (SCSB) et avec modèle interne (SCMI) proposés au chapitre 5, dans le cas d'un processus linéaire, sous forme entrée-sortie, *de retard d=1* :

$$A_p(q)\, y_p\,(k) = B_p(q)\, u\,(k) + A_p(q)\, p(k) = q^{-1}\, B_p'(q)\, u\,(k) + A_p(q)\, p(k)$$

où A_p est un polynôme monique de degré n, et B_p un polynôme de degré m. y_p est la sortie du processus, u la commande, et p est une perturbation additive en sortie du processus. Le modèle est stable et à inverse stable, c'est-à-dire que les zéros de A_p et de B_p sont à l'intérieur du cercle unité.

Les systèmes de commandes étudiés sont tous conçus pour que leur dynamique de poursuite, c'est-à-dire la dynamique de suivi de la consigne r(k), soit la dynamique de référence définie par le modèle :

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

où E et H sont deux polynômes de degré p≥n :

$E(q) = 1 + e_1\, q^{-1} + \ldots + e_p\, q^{-p}$

$H(q) = h_0 + h_1\, q^{-1} + \ldots + h_p\, q^{-p}$

Nous supposons qu'une identification du processus a conduit au modèle linéaire :

$$A(q)\, y\,(k) = B(q)\, u\,(k) = q^{-1}B'(q)\, u\,(k)$$

où A est un polynôme monique de degré n, et B un polynôme de degré m. Les zéros de B sont supposés à l'intérieur du cercle unité.

Enfin, pour rendre compte de l'influence des défauts d'apprentissage du correcteur sur les SCMI, nous considérons que le correcteur n'est généralement pas parfait (parfaitement adapté au modèle).

- D'après le §I.1.1 de cette annexe, l'expression du correcteur-S parfait est (d=1, F=1, B'= q B) :

$$B'(q)\, u(k) = y_r(k+1) + q\left(A(q) - 1\right) y(k)$$

Nous considérons le correcteur suivant :

$$B_c'(q)\, u(k) = y_r(k+1) + q\left(A_c(q) - 1\right) y(k)$$

où A_c est un polynôme monique ; les zéros de B_c sont supposés à l'intérieur du cercle unité (condition nécessaire à la stabilité interne du système de commande).

- D'après le §I.1.2 de cette annexe, le correcteur-D parfait est :

$$B'(q)\, u(k) = H(q)\, r(k) + q\left(A(q) - E(q)\right) y(k).$$

Nous considérons le correcteur suivant :

$$B_c'(q)\, u(k) = H(q)\, r(k) + q\left(A_c(q) - E(q)\right) y(k)$$

Nous étudions les propriétés des systèmes de commande en matière de stabilité, et de performance en poursuite et en régulation :

- pour les SCSB : dans le cas nominal (A_c=A_p, B_c=B_p), et non nominal (A_c≠A_p, B_c≠B_p).
- pour les SCMI : dans le cas nominal (A_p=A, B_p=B), nous étudions l'influence d'une imperfection éventuelle du correcteur (A_c≠A, B_c≠B) ; nous étudions aussi les propriétés du système non nominal, lorsque le correcteur est parfait (A_p≠A, B_p≠B et A_c=A, B_c=B).

II.1. SYSTÈMES DE COMMANDE PAR SIMPLE BOUCLAGE (SCSB).

II.1.1. SCSB utilisant un correcteur-S.

Dans un SCSB, le correcteur-S est mis en cascade avec le processus :

$$B_c'(q)\, u(k) = y_r(k+1) + q\left(A_c(q) - 1\right) y_p(k)$$

où y_p est la sortie mesurée du processus. La sortie de référence y_r est calculée par un modèle de référence qui peut être soit un *modèle de poursuite*, soit un *modèle de ralliement*.

II.1.1.1. Avec un modèle de poursuite.

Le modèle de poursuite s'écrit :

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

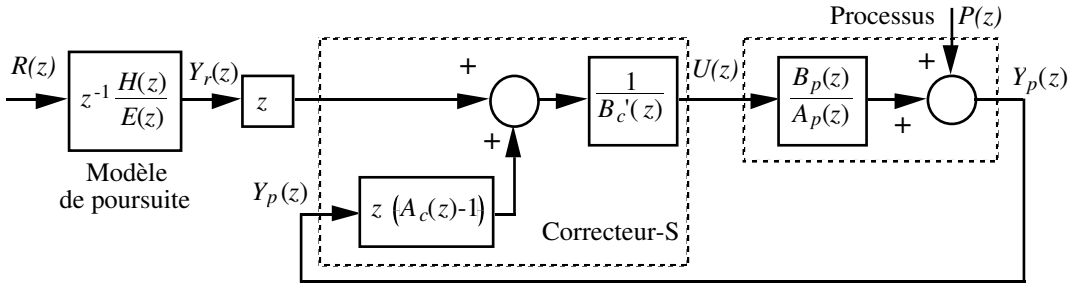

Figure 1.
SCSB avec correcteur-S et modèle de poursuite.

La transformée en z de la commande u de la figure 1, en fonction de la consigne et de la perturbation, est la suivante :

$$U(z) = \frac{A_p(z)}{B_p'(z) + \left(A_p(z)\, B_c'(z) - A_c(z)\, B_p'(z)\right)} \left(\frac{H(z)}{E(z)} R(z) + z\left(A_c(z) - 1\right) P(z)\right)$$

a) Fonctionnement nominal ($A_c = A_p$, $B_c = B_p$) .

$$U(z) = \frac{A_p(z)}{B_p'(z)} \left(\frac{H(z)}{E(z)} R(z) + z\left(A_p(z) - 1\right) P(z)\right)$$

La sortie du processus est :

$$Y_p(z) = z^{-1} \frac{H(z)}{E(z)} R(z) + A_p(z)\, P(z)$$

La dynamique de poursuite du système est évidemment la dynamique du modèle de poursuite. La dynamique de régulation est donnée par le polynôme A_p : le régime permanent est atteint en n pas.
Par définition du correcteur-S, s'il n'y pas de perturbation, il n'y a pas d'erreur statique. Mais soit une perturbation additive de sortie en échelon :

$$P(z) = \frac{P_0}{1 - z^{-1}}$$

Cette perturbation n'est pas rejetée ; elle provoque une erreur statique :

$$E_0 = P_O \sum_{i=0}^{n} a_{ip}$$

De plus, comme l'indique la fonction de transfert de U par rapport à P, si le polynôme B_p possède des zéros négatifs, en réponse à une perturbation la commande oscille à chaque instant. Par rapport à la consigne R, la commande est filtrée par E, et l'effet des éventuels zéros négatifs de B_p est très atténué par le facteur E.

b) Fonctionnement non nominal ($A_c \neq A_p$, $B_c \neq B_p$).

$$Y_p(z) = \frac{1}{B_p'(z) + \left(A_p(z)\ B_c'(z) - A_c(z)\ B_p'(z)\right)} \left(B_p'(z)\ z^{-1}\ \frac{H(z)}{E(z)}\ R(z) + A_p(z)\ B_c'(z)\ P(z) \right)$$

La stabilité du système de commande est fonction de l'inadéquation du correcteur par rapport au processus. Nous allons comparer l'influence de cette inadéquation sur la stabilité de ce système et sur celle du système de commande du paragraphe II.1.1.2 suivant, pour A_p, B_p, A_c et B_c donnés.

II.1.1.2. Avec un modèle de ralliement.

Le modèle de ralliement s'écrit :

$$y_r(k+1) = \left(1 - E(q)\right) y_p(k+1) + H(q)\ r(k)$$

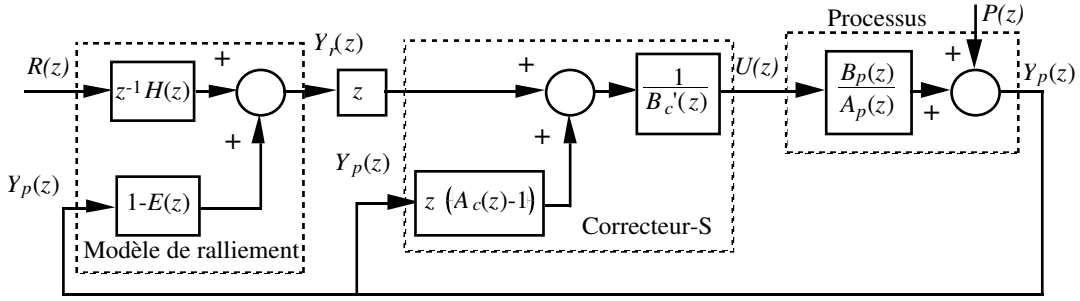

Figure 2.
SCSB avec correcteur-S et modèle de ralliement.

La sortie du correcteur-S de la figure 2 est la suivante :

$$U(z) = \frac{A_p(z)}{E(z)\ B_p'(z) + \left(A_p(z)\ B_c'(z) - A_c(z)\ B_p'(z)\right)} \left(H(z)\ R(z) + z\left(A_c - E(z)\right) P(z) \right)$$

a) Fonctionnement nominal ($A_c = A_p$, $B_c = B_p$) .

$$U(z) = \frac{A_p(z)}{B_p'(z)} \left(\frac{H(z)}{E(z)}\ R(z) + z \left(\frac{A_p(z)}{E(z)} - 1 \right) P(z) \right)$$

La sortie du processus est :

$$Y_p(z) = z^{-1}\ \frac{H(z)}{E(z)}\ R(z) + \frac{A_p(z)}{E(z)}\ P(z)$$

La dynamique de poursuite du système de commande est évidemment encore la dynamique de référence. Grâce au modèle de ralliement, qui possède comme argument la sortie du processus et prend ainsi d'éventuelles perturbations de sortie en considération, la dynamique de régulation est donnée par le polynôme A_p/E : le régime permanent est atteint asymptotiquement. Il existe une erreur statique dont l'amplitude est :

$$E_0 = P_O \frac{\sum_{i=0}^{n} a_{pi}}{\sum_{i=0}^{n} e_i}$$

L'erreur statique est ainsi généralement plus importante que dans le cas précédent[5]. En revanche, la commande est filtrée par E, donc son comportement est satisfaisant en réponse à une perturbation de sortie quel que soit B_p (la commande n'oscille que faiblement si B_p possède des zéros négatifs), ce qui constitue un avantage important de ce système de commande sur le précédent.

b) Fonctionnement non nominal ($A_c \neq A_p$, $B_c \neq B_p$).

$$Y_p(z) = \frac{1}{E(z)B_p'(z) + \left(A_p(z)\ B_c'(z) - A_c(z)\ B_p'(z)\right)} \left(B_p'(z)\ z^{-1}\ H(z)\ R(z) + A_p(z)\ B_c'(z)\ P(z)\right)$$

La stabilité du système de commande est déterminée par les racines du polynôme au dénominateur.

Étude simplifiée de la stabilité.

Nous allons voir que la stabilité de la plupart des systèmes de commande étudiés dépend des racines d'un polynôme D(z) de la forme :

$$D(z) = \varepsilon(z)\ B_1(z) + A_1(z)\ B_2(z) - A_2(z)\ B_1(z)$$

avec, dans le cas du SCSB avec correcteur-S et modèle de poursuite :

$\varepsilon(z) = 1$; $A_1(z) = A_p(z)$; $B_1(z) = B_p'(z)$; $A_2(z) = A_c(z)$; $B_2(z) = B_c'(z)$.

et dans le cas du SCSB avec correcteur-S et modèle de ralliement :

$\varepsilon(z) = E(z)$; $A_1(z) = A_p(z)$; $B_1(z) = B_p'(z)$; $A_2(z) = A_c(z)$; $B_2(z) = B_c'(z)$.

Nous allons montrer sur un exemple que, si $\varepsilon(z)=E(z)\neq 1$, la stabilité du système est plus robuste vis-à-vis d'un écart entre les polynômes A_1 et A_2 et les polynômes B_1 et B_2, que si $\varepsilon(z) = 1$.

Considérons le cas particulier d'un processus du premier ordre, dont le correcteur n'est pas parfait, mais possède un gain statique égal à l'inverse de celui du processus, 1 par exemple :

$$\begin{cases} A_1(q) = 1 + a_1\ q^{-1} \\ B_1(q) = (1 + a_1)\ q^{-1} \end{cases}, \begin{cases} A_2(q) = 1 + a_2\ q^{-1} \\ B_2(q) = (1 + a_2)\ q^{-1} \end{cases}, \varepsilon(q) = 1 + e\ \ q^{-1}$$

où a_1, a_2 et e sont des réels négatifs appartenant à]-1; 0[. La racine du polynôme D(z) s'écrit :

$$\frac{a_2 - a_1 - e\ (1 + a_1)}{1 + a_2} = 1 - \frac{(1 + e\ )(1 + a_1)}{1 + a_2}$$

La valeur de la racine n'est jamais supérieure à 1 ; la condition pour qu'elle soit inférieure à -1 (et donc que le système de commande soit instable) est :

$$2\ \frac{1 + a_2}{1 + a_1} < 1 + e$$

Plus e est petit ($e \leq 0$), moins le système risque d'être instable. En conclusion, pour A_1 et A_2 donnés, le système de commande est d'autant plus stable que E impose une dynamique lente (filtre passe-bas).

---

[5] En effet, la somme des e_i est aussi le produit des racines du polynôme E (monique), toutes de module inférieur à 1 puisque E est stable : cette somme est donc inférieure à 1.

Revenons à la comparaison des SCSB avec modèle de poursuite et modèle de ralliement. Leur stabilité est donc déterminée par D(z), où A_1=A_p, B_1=B_p', A_2=A_c, B_2=B_c' dans les deux cas ; mais pour le système avec modèle de poursuite ε(z)=1, donc e=0, alors que pour le système avec modèle de ralliement e<0 : il vaut donc mieux utiliser un modèle de ralliement du point de vue de la stabilité.
Application numérique : $a_p = a_1 =$ - 0,8 ; $a_c = a_2 =$ - 0,95 .
- SCSB avec modèle de poursuite : le système est instable.

$$2\,\frac{1 - 0{,}95}{1 - 0{,}8} = 0{,}5 < 1$$

- SCSB avec modèle de ralliement : le système est stable dès que $e <$ - 0,5, ce qui permet encore d'accélérer le processus de façon appréciable.

En conclusion, si l'on utilise un correcteur-S dans un SCSB, le modèle de référence du système doit être un modèle de ralliement, qui garantit une meilleure stabilité et un meilleur comportement en réponse à des perturbations de sortie qu'un modèle de poursuite.

II.1.2. SCSB utilisant un correcteur-D.

Dans un SCSB, le correcteur-D est mis en cascade avec le processus :

$$B_c'(q)\,u(k) = H(q)\,r(k) + q\left(A_c(q) - E(q)\right) y_p(k)$$

où y_p est la sortie mesurée du processus. Comme le montre la comparaison de la figure 3 avec la figure 2 (correcteur-S avec modèle de ralliement du §II.1.1.2), les deux systèmes de commande sont rigoureusement équivalents.

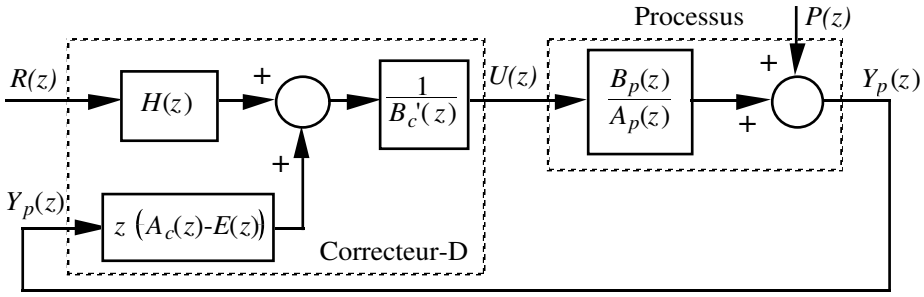

Figure 3.
SCSB avec correcteur-D.

II.2. Systèmes de commande avec modèle interne (SCMI).

Pour les SCMI, nous montrons également l'influence d'un correcteur imparfait ; cette imperfection n'a pas lieu d'être envisagée si les coefficients du correcteur sont calculés à partir de ceux du modèle, mais doit l'être si l'on apprend un réseau correcteur à partir du modèle.

II.2.1. SCMI utilisant un correcteur-S.

Dans un SCMI, le correcteur-S est mis en cascade avec le modèle interne (MI) :

$$B_c'(q)\, u(k) = y_r^*(k+1) + q\left(A_c(q) - 1\right) y(k)$$

où y est la sortie du MI, et y_r^* est la sortie de référence pour le MI. Trois modèles de référence sont *a priori* possibles pour le calcul de la sortie de référence : un *modèle de poursuite à l'extérieur* de la boucle du MI, un *modèle de poursuite à l'intérieur* de celle-ci, ou encore un *modèle de ralliement* pour le MI.

II.2.1.1. Avec un modèle de poursuite extérieur.

L'expression du modèle de poursuite à l'extérieur de la boucle du MI est :

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

L'entrée de référence du correcteur-S est :

$$y_r^*(k+1) = y_r(k+1) - \left(y_p(k) - y(k)\right)$$

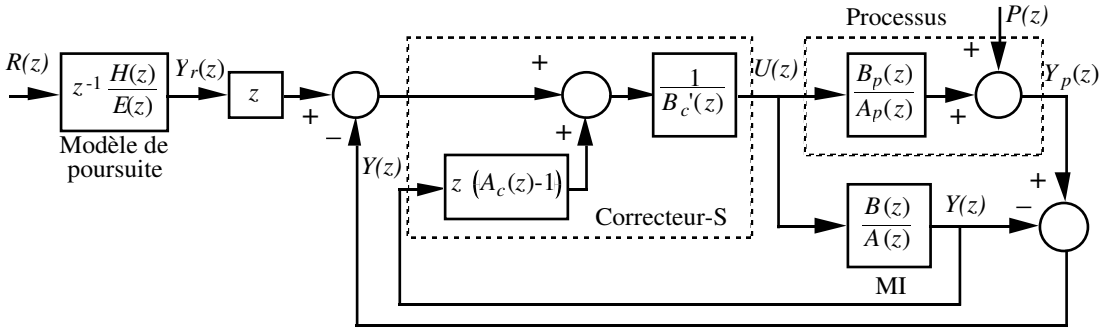

Figure 4.
SCMI avec correcteur-S et modèle de poursuite extérieur.

La commande U(z) de la figure 4 s'écrit, en fonction de la consigne et de la perturbation :

$$U(z) = \frac{A(z)\,A_p(z)}{A_p(z)B'(z) + A_p(z)\left(A(z)B_c'(z) - A_c(z)B'(z)\right) + z^{-1}\left(A(z)B_p'(z) - A_p(z)B'(z)\right)}\left(\frac{H(z)}{E(z)}R(z) - P(z)\right)$$

L'expression ci-dessus fait apparaître au dénominateur l'influence de la désadaptation modèle-processus et de l'imperfection du correcteur.

a) Fonctionnement nominal ($A = A_p$, $B = B_p$) .

Correcteur parfait ($A_c = A$, $B_c = B$) .

$$U(z) = \frac{A_p(z)}{B_p'(z)}\left(\frac{H(z)}{E(z)}\,R(z) - P(z)\right)$$

Faisons apparaître l'intégrateur que réalise implicitement l'organe de commande en exprimant la perturbation P en fonction de Y_p et de U :

$$U(z) = \frac{1}{1 - z^{-1}}\,\frac{A_p(z)}{B_p'(z)}\left(\frac{H(z)}{E(z)}\,R(z) - Y_p(z)\right)$$

Il n'y aura donc pas d'erreur statique. La sortie du processus est :

$$Y_p(z) = z^{-1}\frac{H(z)}{E(z)}R(z) + \left(1 - z^{-1}\right)P(z)$$

La dynamique de régulation est infiniment rapide : en effet, la perturbation est éliminée au bout d'un pas s'il s'agit d'un échelon. La commande oscille fortement à chaque pas d'échantillonnage si le polynôme B_p possède des zéros négatifs.

Correcteur imparfait ($A_c \neq A$, $B_c \neq B$) .

Nous étudions ce cas pour simuler un défaut d'apprentissage du correcteur. La commande est :

$$U(z) = \frac{A_p(z)}{B_p'(z) + \left(A_p(z)\,B_c'(z) - A_c(z)\,B_p'(z)\right)}\left(\frac{H(z)}{E(z)}R(z) - P(z)\right)$$

La stabilité du système de commande dépend du polynôme au dénominateur dont les racines sont étudiées au §II.1.1.2. Dans le cas présent, avec les notations du §II.1.1.2, $\varepsilon(z)=1$ et donc le système de commande risque fort d'être instable si le correcteur n'est pas parfait, et en particulier s'il est adapté à un modèle plus lent (dans l'exemple du §II.1.1.2, un correcteur adapté à un modèle plus lent correspond à $a_2<a_1$).

b) Fonctionnement non nominal ($A \neq A_p$, $B \neq B_p$) ; correcteur parfait ($A_c=A$, $B_c=B$) .

Dans ce cas, la commande s'écrit :

$$U(z) = \frac{A(z)\,A_p(z)}{A_p(z)\,B'(z) + z^{-1}\left(A(z)\,B_p'(z) - A_p(z)\,B'(z)\right)}\left(\frac{H(z)}{E(z)}R(z) - P(z)\right)$$

Même dans le cas non nominal, pourvu que le correcteur soit parfait, l'organe de commande utilisant un MI réalise encore implicitement un intégrateur :

$$U(z) = \frac{1}{1 - z^{-1}}\frac{A(z)}{B'(z)}\left(\frac{H(z)}{E(z)}R(z) - Y_p(z)\right)$$

La performance du système est donc robuste. Le SCSB équivalent est représenté sur la figure 5.

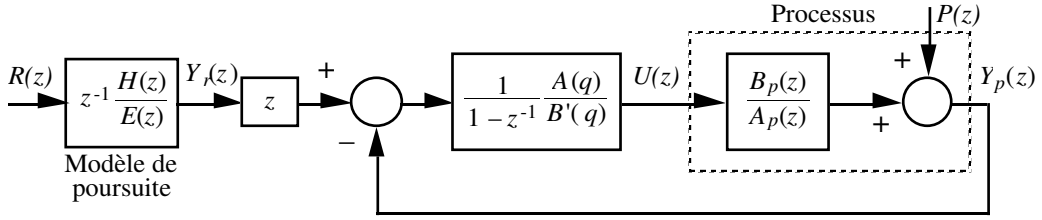

Figure 5.
SCSB équivalent au SCMI avec correcteur-S et modèle de poursuite extérieur
(fonctionnement non nominal, correcteur parfait).

La sortie du processus est :

$$Y_p(z) = \frac{1}{A_p(z)B'(z) + z^{-1}\left(A(z\,)B_p'(z) - A_p(z)B'(z)\right)}\left(z^{-1}A(z)B_p'(z)\frac{H(z)}{E(z)}R(z) + \left(1 - z^{-1}\right)A(z)B_p'(zP(z)\right)$$

On vérifie ainsi qu'il n'y a pas d'erreur statique pour une perturbation de sortie constante.

II.2.1.2. Avec un modèle de poursuite intérieur.

L'expression du modèle de poursuite à l'intérieur de la boucle du MI est :

$$E(q)\, y_r^*(k+1) = H(q)\, r^*(k)$$

où :

$$r^*(k) = r(k) - \left(y_p(k) - y(k)\right)$$

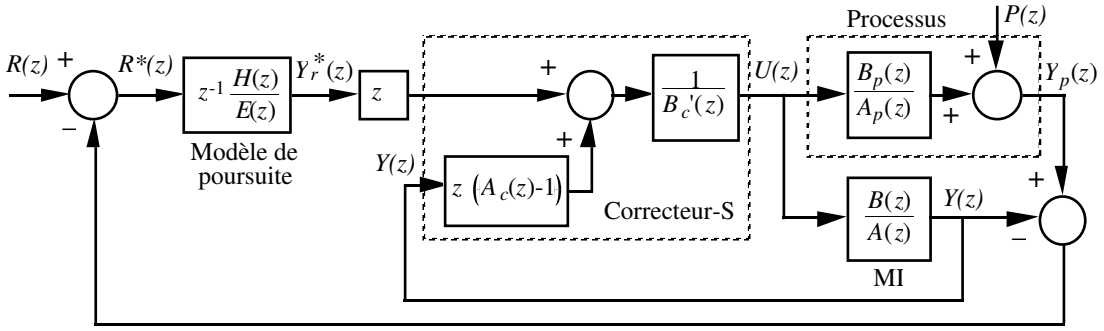

Figure 6.
SCMI avec correcteur-S et modèle de poursuite intérieur.

La sortie du correcteur-S de la figure 6 est la suivante :

$$U(z) = \frac{A(z)\,A_p(z)\,H(z)}{E(z)A_p(z)\left(B'(z) + \left(A(z)B_c'(z) - A_c(z)B'(z)\right)\right) + z^{-1}H(z)\left(A(z)B_p'(z) - A_p(z)B'(z)\right)}\left(R(z) - P(z)\right)$$

a) Fonctionnement nominal ($A = A_p$, $B = B_p$) .

Correcteur parfait ($A_c = A$, $B_c = B$) .

$$U(z) = \frac{A_p(z)}{B_p'(z)}\,\frac{H(z)}{E(z)}\left(R(z) - P(z)\right)$$

La dynamique de régulation est identique à la dynamique de poursuite, et donc à la dynamique de référence. Le dénominateur commun de la commande est E B_p, alors que dans le cas du SCMI avec correcteur-S et modèle de poursuite extérieur (§II.2.1.1), le dénominateur de la fonction de transfert par rapport à R est E B_p, mais le dénominateur de la fonction de transfert par rapport à P est B_p seulement. Si le polynôme B_p possède des zéros négatifs, l'effet en sera atténué pour le présent système.

Exprimons la commande en fonction de l'erreur :

$$U(z) = \frac{H(z)}{E(z) - z^{-1}\,H(z)}\,\frac{A_p(z)}{B_p'(z)}\left(R(z) - Y_p(z)\right)$$

Le premier facteur du dénominateur s'écrit :

$$E(z) - z^{-1}\,H(z) = \sum_{i=0}^{p} e_i\, z^{-i} - \sum_{i=0}^{p} h_i\, z^{-(i+1)}$$

Or le modèle de référence étant choisi de gain statique unité, on a l'égalité :

$$\sum_{i=0}^{p} e_i = \sum_{i=0}^{p} h_i$$

1 est donc racine du polynôme, ce qui nous permet d'écrire le polynôme sous la forme :

$$E(z) - z^{-1} H(z) = \left(1 - z^{-1}\right) Q(z)$$

où Q(z) est un polynôme monique de degré p-1. L'organe de commande réalise encore implicitement un intégrateur :

$$U(z) = \frac{1}{1 - z^{-1}} \frac{H(z)}{Q(z)} \frac{A_p(z)}{B_p'(z)} \left(R(z) - Y_p(z)\right)$$

La sortie du processus est :

$$Y_p(z) = z^{-1} \frac{H(z)}{E(z)} R(z) + \left(1 - z^{-1} \frac{H(z)}{E(z)}\right) P(z)$$

Correcteur imparfait ($A_c \neq A$, $B_c \neq B$) .

Cette éventualité est envisagée pour simuler un défaut d'apprentissage du correcteur. L'expression de la commande est la suivante :

$$U(z) = \frac{A_p(z)}{B_p'(z) + \left(A_p(z)\ B_c'(z) - A_c(z)\ B_p'(z)\right)} \frac{H(z)}{E(z)} \left(R(z) - P(z)\right)$$

Le polynôme apparaissant au dénominateur est le même que pour le système de commande précédent : la stabilité du présent système de commande n'est donc pas non plus robuste vis-à-vis d'une imperfection du correcteur.

b) Fonctionnement non nominal ($A \neq A_p$, $B \neq B_p$) ; correcteur parfait ($A_c = A$, $B_c = B$) .

La commande s'écrit :

$$U(z) = \frac{A(z)\ A_p(z)\ H(z)}{E(z)\ A_p(z)\ B'(z) + z^{-1}\ H(z) \left(A(z)\ B_p'(z) - A_p(z)\ B'(z)\right)} \left(R(z) - P(z)\right)$$

Pourvu que le correcteur soit adapté au modèle, l'organe de commande comprenant un MI réalise encore implicitement un intégrateur, comme le montre la figure 7 :

$$U(z) = \frac{1}{1 - z^{-1}} \frac{H(z)}{Q(z)} \frac{A(z)}{B'(z)} \left(R(z) - Y_p(z)\right)$$

La performance du système est donc robuste.

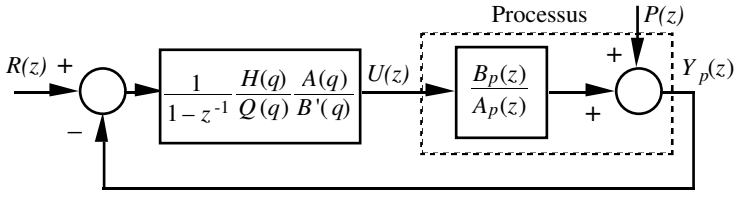

Figure 7.
SCSB équivalent au SCMI avec correcteur-S et modèle de poursuite intérieur
(fonctionnement non nominal, correcteur parfait).

La sortie du processus est :

$$Y_p(z) = \frac{\left(A(z)B_p'(z)z^{-1}H(z)\ R(z) + \left(1 - z^{-1}\right)A_p(z)B'(z)Q(z)\ P(z)\right)}{E(z)A_p(z)B'(z) + z^{-1}\ H(z)\left(A(z)B_p'(z) - A_p(z)B'(z)\right)}$$

On vérifie qu'il n'y a pas d'erreur statique pour une perturbation de sortie constante.

II.2.1.3. Avec un modèle de ralliement.

Le modèle de ralliement rallie la trajectoire du modèle à celle de la consigne décalée. L'expression du modèle de ralliement est :

$$y_r^*(k+1) = \big(1 - E(q)\big)\, y(k+1) + H(q)\; r^*(k)$$

où :

$$r^*(k) = r(k) - \big(y_p(k) - y(k)\big)$$

La sortie du correcteur-S de la figure 8 est la suivante :

$$U(z) = \frac{A(z)\,A_p(z)\,H(z)}{E(z)A_p(z)B'(z) + A_p(z)\big(A(z)B_c'(z) - A_c(z)B'(z)\big) + z^{-1}H(z)\big(A(z)B_p'(z) - A_p(z)B'(z)\big)}\,\big(R(z) - P(z)\big)$$

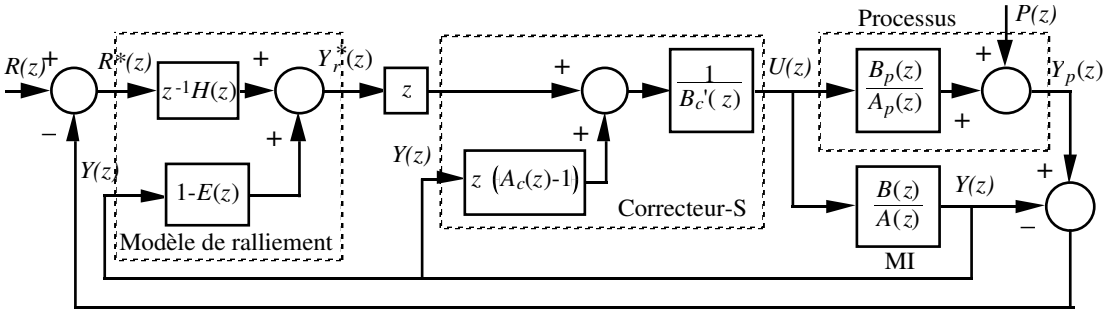

Figure 8
SCMI avec correcteur-S et modèle de ralliement.

a) Fonctionnement nominal ($A = A_p$, $B = B_p$) .

Correcteur parfait ($A_c = A$, $B_c = B$) .

Le système est identique au SCMI avec correcteur-S et modèle de poursuite intérieur (§II.2.1.2).

Correcteur imparfait ($A_c \neq A$, $B_c \neq B$) .

L'expression de la commande est la suivante :

$$U(z) = \frac{A_p(z)\,H(z)}{E(z)\,B_p'(z) + \big(A_p(z)\,B_c'(z) - A_c(z)\,B_p'(z)\big)}\,\big(R(z) - P(z)\big)$$

Le polynôme apparaissant au dénominateur fait intervenir E(z) (soit $\varepsilon(z) \neq 1$ avec les notations du §II.1.1.2) : la stabilité du présent système de commande est donc plus robuste vis-à-vis d'une imperfection du correcteur, que celle des deux systèmes précédents (où $\varepsilon(z) = 1$).

b) Fonctionnement non nominal ($A \neq A_p$, $B \neq B_p$); correcteur parfait ($A_c = A$, $B_c = B$) .

Le fonctionnement du système est identique à celui du SCMI avec correcteur-S et modèle de poursuite intérieur (§II.2.1.2). Ses propriétés de performance robuste sont les mêmes.

II.2.2. SCMI utilisant un correcteur-D.

L'entrée de consigne du correcteur-D est :

$$r^*(k) = r(k) - \big(y_p(k) - y(k)\big)$$

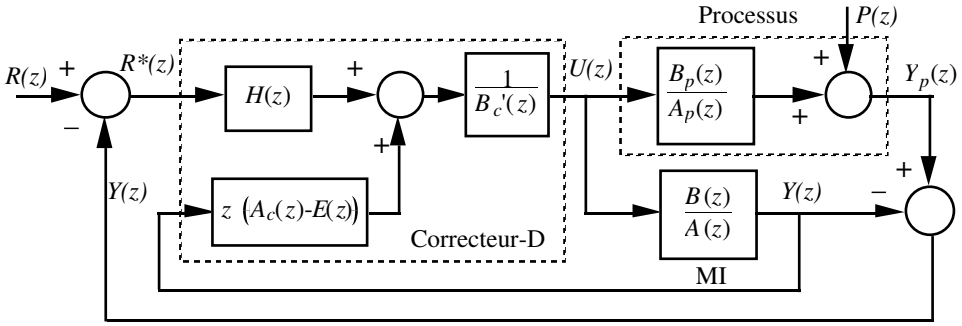

Figure 9.
SCMI avec correcteur-D.

La sortie du correcteur-D du schéma de la figure 9 est la suivante :

$$U(z) = \frac{A(z)\,A_p(z)\,H(z)}{E(z\,)A_p(z)B'(z) + A_p(z)\big(A(z)B_c'(z) - A_c(z)B'(z)\big) + z^{-1}H(z)\big(A_p(z)B'(z) - A(z)B_p'(z)\big)}\big(R(z) - P(z)\big)$$

a) Fonctionnement nominal ($A = A_p$, $B = B_p$).

Correcteur parfait ($A_c = A$, $B_c = B$) .

$$U(z) = \frac{A_p(z)}{B_p'(z)}\,\frac{H(z)}{E(z)}\big(R(z) - P(z)\big)$$

En faisant apparaître l'intégrateur implicite :

$$U(z) = \frac{1}{1 - z^{-1}}\,\frac{H(z)}{Q(z)}\,\frac{A_p(z)}{B_p'(z)}\big(R(z) - Y_p(z)\big)$$

La sortie du processus est :

$$Y_p(z) = z^{-1}\,\frac{H(z)}{E(z)}\,R(z) + \left(1 - z^{-1}\frac{H(z)}{E(z)}\right)P(z)$$

La dynamique de régulation est la même que la dynamique de poursuite : elle donnée par le polynôme de référence 1/E. Il n'y a pas d'erreur statique pour une perturbation de sortie constante puisque le modèle de référence est de gain unité. Le comportement de la commande est toujours satisfaisant quels que soient les zéros du processus (la commande est filtrée par E).

Correcteur imparfait ($A_c \neq A$, $B_c \neq B$) .

L'expression de la commande est la suivante :

$$U(z) = \frac{A_p(z)\,H(z)}{E(z)\,B_p'(z) + \big(A_p(z)\,B_c'(z) - A_c(z)\,B_p'(z)\big)}\big(R(z) - P(z)\big)$$

Dans ce cas, le système de commande est équivalent au système avec correcteur-S et modèle de ralliement. Il est donc aussi relativement robuste vis-à-vis d'une imperfection du correcteur.

b) Fonctionnement non nominal ($A \neq A_p$, $B \neq B_p$) ; correcteur parfait ($A_c = A$, $B_c = B$) .

Dans ce cas, la commande s'écrit :

$$U(z) = \frac{A(z)\,A_p(z)\,H(z)}{E(z)\,A_p(z)\,B'(z) + z^{-1}\,H(z)\,\big(A(z)B_p'(z) - A_p(z)\,B'(z)\big)}\big(R(z) - P(z)\big)$$

C'est la même expression que dans le cas précédent : on retrouve les propriétés de la commande avec correcteur-S et modèle de ralliement.

CONCLUSION.

Presque tous les systèmes de commande étudiés dans cette annexe sont proposés dans la littérature, mais rarement comparés objectivement. L'intérêt de ce travail est : d'une part, de les comparer sur la base de critères qui sont stabilité et performance du système nominal, et robustesse de cette stabilité et de cette performance ; d'autre part, d'indiquer lesquels de ces systèmes doivent être évités, et lesquels doivent être utilisés pour une commande neuronale. Récapitulons les conclusions de cette étude :

- Les systèmes de commande par simple bouclage (SCSB) ou avec modèle interne (SCMI) utilisant un correcteur-S et un modèle de poursuite extérieur à la boucle de retour d'état et, le cas échéant, extérieur à la boucle du modèle interne, ont la caractéristique d'une dynamique de régulation non maîtrisée, et d'une stabilité peu robuste. Les systèmes des paragraphes II.1.1.1 et II.2.1.1 utilisant un tel modèle de poursuite doivent donc être évités.
- Les SCMI présentent l'avantage sur les SCSB d'éliminer les erreurs statiques dues à toute perturbation de sortie en fonctionnement nominal, et de présenter une grande robustesse du niveau de performance vis-à-vis d'écarts par rapport au système nominal (erreur statique nulle pour le système non nominal).
- Un SCMI utilisant un correcteur-S doit comprendre un modèle de ralliement (et non un modèle de poursuite intérieur à la boucle du modèle interne), qui rend la stabilité du système plus robuste vis-à-vis d'une éventuelle imperfection du correcteur.
- Les SCMI utilisant un correcteur-D, ou un correcteur-S et un modèle de ralliement, ont des propriétés voisines. Cependant, le second doit être préféré si l'on utilise des réseaux de neurones, car changer de dynamique de poursuite si nécessaire (par exemple pour stabiliser le processus) ne demande pas de nouvel apprentissage, mais seulement de modifier le modèle de ralliement.