

Mathématique et automatique : de la boucle ouverte à la boucle fermée

Maitine Bergounioux

Laboratoire MAPMO - UMR 6628

Université d'Orléans

BP 6759

F-45067 Orléans Cedex 2, France

`Maitine.Bergounioux@labomath.univ-orleans.fr`

Mathématique et automatique : de la boucle ouverte à la boucle fermée

Maitine bergounioux

Laboratoire MAPMO - UMR 6628

Université d'Orléans BP 6759

F-45067 Orléans Cedex 2

Maitine.Bergounioux@labomath.univ-orleans.fr

1^{er} mars 2003

Je suis professeur de mathématiques à l'université d'Orléans. J'enseigne essentiellement aux niveaux Bac +3, Bac +4, l'analyse numérique, l'analyse de Fourier et le traitement du signal. En DESS, j'enseigne l'optimisation en dimension finie et le contrôle optimal des systèmes linéaires gouvernés par des équations différentielles. Mon domaine de recherche porte sur le contrôle optimal des systèmes distribués (gouvernés par des EDPs ou des inéquations variationnelles), tant du point de vue numérique que théorique.

Résumé

Le but de cette intervention est de cerner un peu les liens entre les mathématiciens et les automaticiens. Après une présentation des différents concepts utilisés par les deux communautés je me concentre un peu plus sur l'optimisation et la commande optimale en donnant quelques outils majeurs et des exemples. J'essaie ensuite de montrer comment ce domaine utilise et a besoin des deux compétences pour progresser.

Mots-clé : Boucle ouverte, boucle fermée, commande optimale, commande prédictive

1 Un peu de sémantique . . .

Le but de cette intervention est de cerner un peu les liens entre les mathématiciens et les automaticiens. Par mathématiciens, j’entends ceux qui travaillent dans les domaines du contrôle, que ce soit la contrôlabilité ou le contrôle optimal. Un des premiers à s’être intéressé à ce champ (à ma connaissance) et qui fut, du point de vue mathématique, celui qui a donné un cadre général, abstrait (parfois trop) et théorique au concept de contrôle a été Jacques-Louis Lions, relayé ensuite par une foule d’élèves et de collaborateurs. Toutefois, si la qualité des travaux effectués depuis trente ans est indéniable, il me semble que si beaucoup de résultats publiés sont d’un grand intérêt du point de vue mathématique (rigueur, “belles mathématiques”), l’applicabilité de ces résultats est souvent faible et il convient pour un mathématicien dit “appliqué” de revenir à la réalité et de voir comment ses “beaux” résultats peuvent être utilisés en pratique. C’est en ce sens que le rapprochement avec les automaticiens est fondamental car l’automatique est en prise avec le réel (il faut en dernière instance créer des automates ou des robots qui fonctionnent vraiment).

Prenons un instant un dictionnaire [5], pour donner quelques définitions et les commenter :

- **Contrôler:**
 - ◆ Vérifier (voir *examiner, inspecter, vérifier*) . . .
 - ◆ Maîtriser, dominer .

- **Commander :**
 - ◆ Exercer son autorité (voir *contraindre, obliger*) . . .
 - ◆(XXe) Prescrire de manière autoritaire . . .
 - ◆ *Techn* Faire fonctionner :
ce mécanisme commande l’ouverture des portes

Même si le mot “commander” a une origine anglo-saxonne, il n’en reste pas moins qu’il y a en français une subtile différence entre les deux termes qui me semble bien refléter les pratiques des deux communautés. “Commander” a un sens **actif**, “contrôler ” un sens plus **passif**. Il est différent de contrôler le débit d’un ruisseau (pour empêcher qu’il ne déborde) que de commander ce débit, c’est-à-dire d’agir pour que le ruisseau aille où l’on veut qu’il aille.

La différence peut sembler très mince mais elle illustre bien la philosophie de chacune des deux communautés. Le mathématicien contrôle, c'est-à-dire cherche un contrôle (optimal ou pas) permettant d'atteindre un but puis, il "laisse faire" : c'est la **boucle ouverte**. On considère un système donné par son équation d'état et un ensemble de contrôles. On cherche un contrôle qui stabilise le système ou un contrôle optimal (ou tout autre type de contrôle) qui ne dépend pas a priori de l'état du système à un moment quelconque. Ce type de stratégie peut se résumer par le schéma suivant:

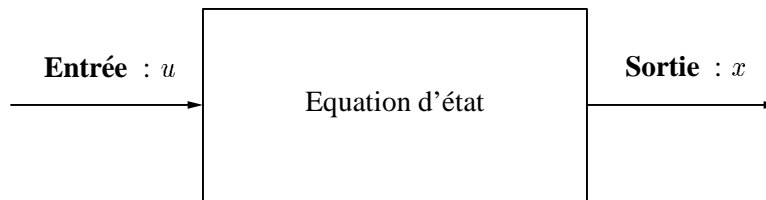


Figure 1 : Boucle ouverte

L'automaticien, lui, cherche un contrôle (ou commande) **impose une loi de commande** (c'est en ce sens qu'il est actif) pour réinjecter la sortie dans le système : c'est la **boucle fermée**. On considère toujours un système donné par son équation d'état et un ensemble de contrôles. On applique le contrôle au système. L'équation d'état retourne une fonction d'état (contrôlée). On introduit alors une loi de commande qui permet d'exprimer la fonction de contrôle précisément en fonction de cet état. L'état du système est pris en compte à chaque instant pour déterminer "en temps réel" la commande. Le contrôle est alors appelé contrôle **feedback**. Cette stratégie est schématisée par la figure suivante.

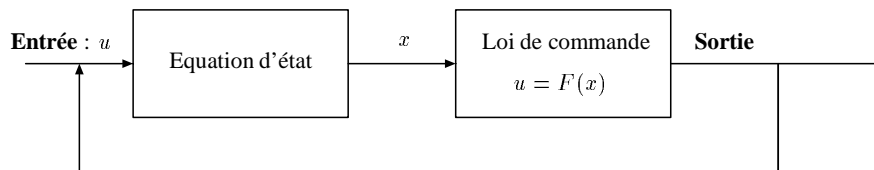


Figure 2 : Boucle fermée

Quiconque a pris une douche a expérimenté la boucle ouverte et la boucle fermée : dans le premier cas, on choisit "au mieux" la température de l'eau en agissant sur le robinet et on n'y touche plus ensuite; dans le second cas, on

adapte en permanence pour pallier les défaillances temporaires de la chaudière et garder ainsi une température constante.

Il est d'ailleurs significatif que de plus en plus de mathématiciens parlent de "contrôles actifs". Je ne vais pas faire le procès de l'une ou l'autre communauté (pour moi les termes "actif" ou "passif" ne sont pas péjoratifs) mais essayer de comparer les différents points de vue et voir comment les rapprocher.

Revenons au dictionnaire pour en savoir plus sur nos deux champs :

Mathématiques: Ensemble des sciences qui ont pour objet la quantité et l'ordre.

Automatique: Qui, une fois mis en mouvement, fonctionne de lui-même.

On ne peut pas dire que cela nous aide beaucoup !!!

2 Dimension finie/ dimension infinie

2.1 La notion de système

La notion de système bien connue des automaticiens et des spécialistes du traitement du signal est peu utilisée par les mathématiciens si ce n'est de manière abstraite: un système est modélisé par un espace vectoriel (normé) d'entrée (*Input*) X et un espace de sortie (*Output*) Y et par la donnée d'une application (en général linéaire) $\mathcal{A} : X \rightarrow Y$ (cf. [8] par exemple). Cela permet ensuite de définir un **filtre** comme étant un système linéaire, causal et invariant. Cette définition n'est guère utilisée que pour les propriétés des espaces; ceux-ci sont en général de dimension infinie, quand les équations d'état sont des équations aux dérivées partielles (EDPs). Or une modélisation fine implique de plus en plus l'usage des EDPs, y compris en automatique [11] par exemple. Une semi-discrétisation (en espace) permet en général de se ramener à des espaces de dimension finie où les équations sont alors des équations différentielles ordinaires (EDO). Il est alors plus aisé de parler de filtre, de "boîte noire" et de boucle fermée.

2.2 La fonction de transfert

Un autre outil spécifique à l'automatique et au traitement du signal est la notion de **fonction de transfert**, y compris en dimension infinie. Là encore, c'est très lié à l'usage de la boucle fermée. C'est en général grâce à ces fonctions que l'on peut définir une loi de commande. La notion de stabilité (asymptotique ou pas) est étudiée à l'aide des pôles de cette fonction. Un mathématicien étudiera la stabilité ou la stabilisation d'un système, "directement" par une analyse de sensibilité de l'équation d'état ou par des estimations des solutions. Les résultats sont souvent comparables : les contrôles/commandes obtenus sont souvent des contrôles feedback. Tout est question de méthode et de ... culture.

2.3 A propos de la contrôlabilité

Je n'aborderai pas très longuement la question de la contrôlabilité et les notions qui lui sont liées (observabilité) par manque de compétence. Une fois de plus, je constate que si le but est le même, les méthodes diffèrent. L'utilisation de la matrice de contrôlabilité en dimension finie provient implicitement de la fonction de transfert. En dimension infinie, c'est encore plus clair. Les mathématiciens utilisent plutôt des outils d'analyse de sensibilité, de principe d'unicité (Théorème de Mizohata par exemple) et font une grande distinction entre la contrôlabilité exacte et la contrôlabilité approchée. Du point de vue théorique il est important d'avoir des résultats de contrôlabilité exacte afin de bien décrire les ensembles commandables et atteignables. Toutefois, du point de vue numérique la contrôlabilité approchée est largement suffisante. En effet on ne peut jamais, lors d'un calcul, atteindre exactement un objectif.

Quand on ne peut pas atteindre un objectif donné, on peut restreindre ses prétentions et essayer de s'en approcher le plus possible. Ainsi on transforme un problème de contrôlabilité, d'atteignabilité ou d'identification de paramètres en un problème d'optimisation. On se contente en effet de minimiser (par exemple) la distance entre la trajectoire commandée et la trajectoire désirée puisqu'on ne peut pas faire mieux. Si la solution obtenue donne une distance nulle ou petite, tant mieux !! Cependant, dans la majorité des cas

on n'atteint pas une valeur du critère nulle : en effet les données sont souvent bruitées, les calculs ne sont pas exacts, etc...

C'est donc, via la minimisation au sens des moindres carrés que nous allons étudier des problèmes de contrôle/ commande optimale.

3 Un peu d'optimisation

Comme je l'ai mentionné ci-dessus, on peut essayer de commander un système pour l'amener à un état désiré en un temps donné (ou minimal) mais on peut aussi vouloir le faire de manière optimale. Avant de parler de commande optimale je vais donc rappeler quelques principes d'optimisation [2].

L'optimisation (c'est-à-dire les techniques permettant de chercher les minima ou les maxima de fonctions ou de fonctionnelles) intervient dans pratiquement tous les processus de modélisation actuels. Qu'il s'agisse de problèmes directs (ajustement de données, contrôle optimal, résolution de systèmes linéaires par moindres carrés, etc ...) ou inverses (identification de paramètres, contrôle de frontières libres etc..), il est rare qu'un problème d'optimisation plus ou moins complexe n'intervienne pas à un stade donné de la modélisation et/ou de la simulation.

Je vais donner un aperçu des techniques d'optimisation en dimension finie. En réalité, on retrouve les mêmes techniques en dimension infinie (c'est-à-dire quand des espaces fonctionnels remplacent des espaces de dimension finie de type \mathbb{R}^n). Les difficultés mathématiques sont plus importantes et pour les besoins de cet exposé je préfère me limiter à un cas simple. Je donnerai un exemple en dimension infinie dans la section suivante. De toutes façons, en dernière instance, si on veut résoudre numériquement le problème (ce qui est tout de même l'objectif) il faut discrétiser le problème et le ramener à un problème de dimension finie. Je ne rentrerai pas dans la discussion de savoir s'il faut d'abord appliquer les algorithmes d'optimisation en dimension infinie puis discrétiser ou d'abord discrétiser et appliquer les algorithmes d'optimisation sur le problème discret. En fait, les deux opérations ne sont pas équivalentes

(voir [1]).

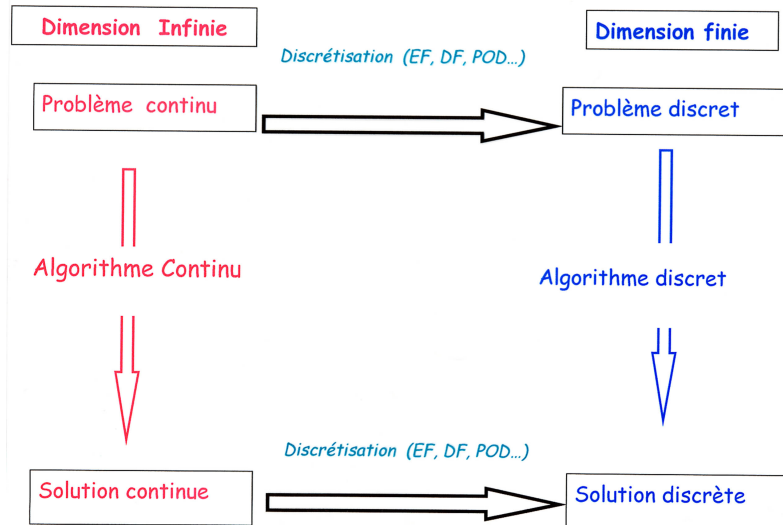


Figure 3

Un problème d'optimisation s'écrit sous la forme générale suivante

$$(\mathcal{P}) \quad \begin{cases} \min J(x) \\ g(x) \leq 0, \\ h(x) = 0, \\ x \in \mathbb{R}^n \end{cases}$$

où

- $J : \mathbb{R}^n \rightarrow \mathbb{R}$ est une fonction de plusieurs variables ($x = (x_1, \dots, x_n)$) à **valeurs réelles**. Cette fonction (que l'on minimise) est appelée indifféremment fonction **coût**, **objectif** ou **critère**.
- $g : \mathbb{R}^n \rightarrow \mathbb{R}^p$ est une fonction de plusieurs variables $x \in \mathbb{R}^n$ à valeurs dans \mathbb{R}^p : elle a p composantes et on peut écrire

$$g(x) = (g_1(x), \dots, g_p(x)) ,$$

chaque fonction g_i étant définie sur \mathbb{R}^n et à valeurs dans \mathbb{R} . La fonction g représente les **contraintes en inégalité**. La notation $g(x) \leq 0$ signifie qu'on considère les inégalités composante par composante :

$$g(x) \leq 0 \stackrel{\text{def}}{\iff} \forall i = 1, \dots, p \quad g_i(x) \leq 0 .$$

- $h : \mathbb{R}^n \rightarrow \mathbb{R}^q$ est une fonction de plusieurs variables $x \in \mathbb{R}^n$ à valeurs dans \mathbb{R}^q : elle a q composantes et on peut écrire

$$h(x) = (h_1(x), \dots, h_q(x)) ,$$

chaque fonction h_i étant définie sur \mathbb{R}^n et à valeurs dans \mathbb{R} . La fonction h représente les **contraintes en égalité** .

Soit \mathcal{C} l'ensemble des contraintes, c'est-à-dire par exemple dans le cas précédent

$$\mathcal{C} = \{ x \in \mathbb{R}^n \mid g(x) \leq 0 , h(x) = 0 \} .$$

On suppose que \mathcal{C} est non vide; un élément x de \mathcal{C} sera dit **réalisable**. On peut imaginer des ensembles de contraintes plus “pathologiques” mais celui que nous avons choisi décrit bien la grande majorité des situations. On dit que $x^* \in \mathcal{C}$ réalise un **minimum local** de f sur \mathcal{C} si on peut trouver une boule centrée en $x^* : \mathcal{B}(x^*)$ telle que

$$\forall x \in \mathcal{B}(x^*) \cap \mathcal{C} \quad f(x^*) \leq f(x) .$$

On dit que $x^* \in \mathcal{C}$ réalise un **minimum global** de f sur \mathcal{C} si

$$\forall x \in \mathcal{C} \quad f(x^*) \leq f(x) .$$

En pratique, il est très difficile de trouver un optimum global: la plupart des outils et algorithmes déterministes dont nous disposons fournissent des optima locaux. On peut toutefois citer l'algorithme du recuit simulé (stochastique) qui donne des optima globaux. La situation la plus favorable reste le cas où les données sont convexes. Ce n'est évidemment pas le cas en pratique mais la majorité des algorithmes modernes sont basés sur une convexification locale des problèmes.

La démarche habituelle consiste (en supposant que les fonctions sont de classe \mathcal{C}^1) à exhiber des conditions d'optimalité du premier ordre (en général seulement nécessaires), voire du second ordre si les données sont assez régulières. La plus célèbre de ces conditions est le système d'optimalité de Kuhn-Karush-Tucker (KKT) que j'explicite ci-dessous.

Soit x^* une solution du problème (\mathcal{P}) (vérifiant une hypothèse de qualification que je n'expliciterai pas). Alors il existe $\lambda^* = (\lambda_1^*, \dots, \lambda_p^*) \in \mathbb{R}^p$ et $\mu^* = (\mu_1^*, \dots, \mu_q^*) \in \mathbb{R}^q$ tels que

$$\forall j \in \{1, \dots, q\} \quad \mu_j^* \geq 0, \quad (3.1a)$$

$$h(x^*) = 0, \quad g(x^*) \leq 0, \quad \text{réalisabilité} \quad (3.1b)$$

$$\forall j \in \{1, \dots, q\} \quad \mu_j^* g_j(x^*) = 0, \quad \text{complémentarité} \quad (3.1c)$$

$$\nabla J(x^*) + \sum_{i=1}^p \lambda_i^* \nabla h_i(x^*) + \sum_{j=1}^q \mu_j^* \nabla g_j(x^*) = 0. \quad (3.1d)$$

On appelle **Lagrangien** du problème (\mathcal{P}) la fonction définie sur $\mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q$ par

$$\mathcal{L}(x, \lambda, \mu) = J(x) + \sum_{i=1}^p \lambda_i h_i(x) + \sum_{j=1}^q \mu_j g_j(x),$$

λ^* et μ^* sont les **multiplicateurs de Lagrange**. Ces conditions d'optimalité se retrouvent sous une forme à peine différente en dimension infinie. Ce sont elles que nous résolvons en réalité (et pas le problème initial). Pratiquement tous les algorithmes déterministes sont basés sur ces conditions et n'ont pour seul but que leur résolution. On voit donc que les solutions obtenues sont des points **stationnaires** et pas nécessairement des optima, puisque ces conditions ne sont pas suffisantes (sauf dans le cas convexe). Néanmoins, les méthodes issues de ces conditions (méthodes dites **Lagrangiennes** car elles utilisent le lagrangien) sont très utilisées. Couplées avec la méthode de Newton ou des méthodes quasi-Newton pour résoudre (3.1d), elles forment la classe des algorithmes de **Lagrange-Newton** qui sont parmi les plus performants. On y trouve par exemple la fameuse méthode de Levenberg-Marquardt.

4 La commande optimale

L'optimisation en théorie de la commande (ou du contrôle) occupe une place à part car la problématique diffère quelque peu. Si on veut commander un système, on se donne **une loi de commande** et on fait une boucle fermée.

A priori, on peut penser que l'usage d'une loi de commande impose le contrôle et de fait induit son unicité.

En commande optimale on cherche une commande qui minimise un critère donné. Rappelons en brièvement le principe (en dimension finie). On se donne tout d'abord un espace (fonctionnel) de contrôles par exemple :

$$\mathcal{U}_p = \bigcup_{T>0} \{ u : \mathbb{R} \rightarrow \mathbb{R}^p \mid u \text{ est intégrable sur } [0, T] \}$$

et un sous-ensemble $U_{ad} \subset \mathcal{U}_p$ (non vide et fermé) de contrôles admissibles : c'est l'ensemble des **contraintes sur le contrôle**.

On considère également un système dynamique dont l'état x est décrit par une équation de la forme $x = \mathcal{F}(u, t)$. Cette équation peut être une équation aux différences, une équation différentielle (ou un système différentiel), un système d'équations aux dérivées partielles, etc... Elle peut être linéaire ou pas. Pour fixer les idées, nous allons étudier la situation suivante: l'état du système $x : [0, +\infty[\rightarrow \mathbb{R}^n$ est donné par une équation différentielle ordinaire (EDO) de la forme

$$\begin{cases} \frac{dx}{dt}(t) = f(t, x(t), u(t)) & t \in]0, +\infty[, \\ x(0) = x_0 \text{ fixé dans } \mathbb{R}^n, \end{cases} \quad (4.2)$$

et on supposera que l'équation (4.2) a une solution unique notée $x = x[u](\cdot)$. On se donne alors une fonctionnelle **coût** J et on cherche à résoudre

$$(\mathcal{P}) \quad \begin{cases} \min J(x, u) \\ x = x[u] & \text{Equation d'état} \\ u \in U_{ad} & \text{Contraintes sur le contrôle.} \end{cases}$$

Très souvent, il apparaît aussi des contraintes sur la variable d'état de la forme

$$x \in \mathcal{K}$$

où \mathcal{K} est un sous-ensemble non vide et fermé de l'espace d'état.

Le choix de la fonctionnelle J se fait en fonction du problème considéré. S'il s'agit d'un problème d'économie, J pourra représenter l'opposé d'un bénéfice (car en économie on maximise). S'il s'agit d'un problème de mécanique, J

pourra être une énergie consommée (par exemple le carburant pour déplacer un véhicule). Si la fonctionnelle J est de la forme

$$J(x, u) = \int_0^T \Phi(x(t), u(t)) dt ,$$

on dira qu'on a un problème de contrôle optimal à **horizon fini** (T). Si, elle est de la forme

$$J(u) = \int_0^t ds ,$$

on aura un problème de **temps minimal**.

On pourrait penser qu'imposer une loi de commande dans ce contexte risque d'empêcher l'existence de la solution. En fait la situation est paradoxale. J'en donne un exemple, et je me place du point de vue mathématique.

Considérons un problème tout à fait académique pour simplifier, proposé par J.L. Lions [10]. On veut que la température d'une pièce soit de 20°C le temps d'une conférence. On suppose qu'on est en régime stationnaire (la journée est avancée) et au début de la conférence la pièce est déjà à 20°C. Il suffit donc de maintenir la température de la pièce de manière uniforme. C'est le principe du thermostat. Si on simplifie le modèle, l'équation d'état sera l'équation de la chaleur stationnaire, la fonction coût décrira la différence entre la température au point x et la température désirée, et la commande sera le flux de chaleur dispensé par les radiateurs situés (c'est le cas habituellement) au bord de la pièce. Les personnes présentes dans la pièce constituent aussi une source de chaleur connue (distribuée) mais pas contrôlable. Finalement, en termes mathématiques, si Ω désigne la pièce et $\partial\Omega$ son bord (murs, plancher, plafond) le problème s'écrit

$$(\mathcal{P}) \quad \begin{cases} \min J(\theta, u) \stackrel{\text{def}}{=} \int_{\Omega} |\theta(x) - \theta_o|^2 dx + \frac{\alpha}{2} \int_{\partial\Omega} |u(s)|^2 ds \\ -\Delta\theta = f \text{ dans } \Omega \\ \frac{\partial\theta}{\partial n} = u \text{ sur } \partial\Omega \\ u \in U_{ad} , \end{cases}$$

où $\frac{\partial\theta}{\partial n}$ est la dérivée normale de u sur le bord $\partial\Omega$. Ici f est la source de chaleur distribuée et θ la température (état) et $\theta_o = 20$. Le terme en u dans

J est là pour moduler l'influence du contrôle : on ne chauffe pas à n'importe quel prix. L'ensemble U_{ad} est un ensemble de contrôles admissibles (on veut chauffer mais sans faire exploser la chaudière). Nous avons un problème de contrôle frontière. Ce problème a une solution unique u^* qui est caractérisée par le système d'optimalité suivant (qui est l'équivalent en dimension infinie des conditions de KKT évoquées précédemment) :

$$(\mathcal{S}) \quad \begin{cases} -\Delta\theta^* = f \text{ dans } \Omega, & \frac{\partial\theta^*}{\partial n} = u^* \text{ sur } \partial\Omega & \text{(Equation d'état)}, \\ -\Delta p^* = \theta^* - \theta_o \text{ dans } \Omega, & -\frac{\partial p^*}{\partial n} = 0 \text{ sur } \partial\Omega & \text{(Equation adjointe)}, \\ u^* = -\Pi_{U_{ad}}(p^*), \end{cases}$$

où $\Pi_{U_{ad}}$ désigne la projection de $L^2(\Omega)$ sur U_{ad} . On constate donc que u^* est une fonction de θ^* via l'état adjoint p^* et nous avons donc affaire à un contrôle feedback. De ce fait, nous avons construit une loi de commande "optimale" qui permet de gérer le système en boucle fermée alors qu'il était a priori en boucle ouverte.

La démarche décrite ci-dessous est la démarche standard du mathématicien qui fait du contrôle optimal.

- Modéliser et formuler le problème
- Préciser le cadre fonctionnel (ce que je n'ai pas fait par souci de simplification)
- Prouver l'existence d'une solution
- Discuter l'unicité, quitte à rajouter des contraintes
- Caractériser la solution par un système d'optimalité ou par un principe de Pontryagin
- Utiliser cette caractérisation pour calculer explicitement ou numériquement la solution.

A aucun moment, dans cette démarche, ne se pose la question de la boucle fermée. Le mathématicien travaille essentiellement en boucle ouverte. Ceci étant, comme le montre l'exemple précédent, on peut souvent récupérer un contrôle feedback et le réinjecter ensuite dans le système. Le contrôleur mathématicien est comme M. Jourdain : il fait de la boucle fermée sans le savoir...

5 Commande prédictive

Il est tout de même au moins un domaine où mathématiciens et automaticiens peuvent se retrouver et parler à peu près le même langage et où, si j'ose dire, boucle ouverte et boucle fermée cohabitent. Je parle de celui que je connais un peu, celui de la commande prédictive ([3, 4] par exemple).

5.1 Principe de la commande prédictive

Les techniques d'optimisation ont un rôle fondamental dans l'industrie, non seulement en recherche opérationnelle mais dans des processus industriels dynamiques. Dans beaucoup de cas, on résout un problème de contrôle optimal "off-line", c'est-à-dire avant que tout processus ne commence. C'est là qu'intervient la boucle ouverte.

En pratique, toutefois, les trajectoires "contrôlées" par une optimisation "off-line" sont d'une utilisation limitée: en effet le procédé réel ne coïncide pas complètement avec le modèle mathématique, et de plus est sujet à bruitage. Il est donc très intéressant du point de vue pratique de pouvoir construire des contrôles feedback (boucle fermée donc) basés sur de l'optimisation. Comme on ne peut évidemment pas les calculer à l'avance pour toutes les perturbations possibles, il faut donc le faire **en temps réel**.

L'idée des **modèles de commande prédictive** (MPC) est la suivante: on détermine un contrôle au temps t_o en résolvant un problème de contrôle optimal sur une horizon de **prédiction**: $[t_o, t_o + T]$. On n'utilise la commande trouvée que pour un temps voisin de t_o , et à l'instant $[t_o + \delta, t_o + T + \delta]$ on résout un nouveau problème de contrôle optimal. On a utilisé une fenêtre "glissante" en quelque sorte. On obtient donc une suite de problèmes d'optimisation formulés et résolus en temps réel ce qui autorise la correction rapide des perturbations. Les modèles de commande prédictive linéaire sont aujourd'hui largement utilisés (voir par exemple [7] ou [9]).

Pour des procédés longs ou périodiques, on peut espérer que des modèles **non linéaires** basés sur les même principes que précédemment, puissent donner une "image" plus précise du comportement des systèmes qu'avec les modèles linéaires. C'est ainsi que sont apparus des modèles de commande

prédictive non linéaires (NMPC) largement étudiés du point de vue théorique.

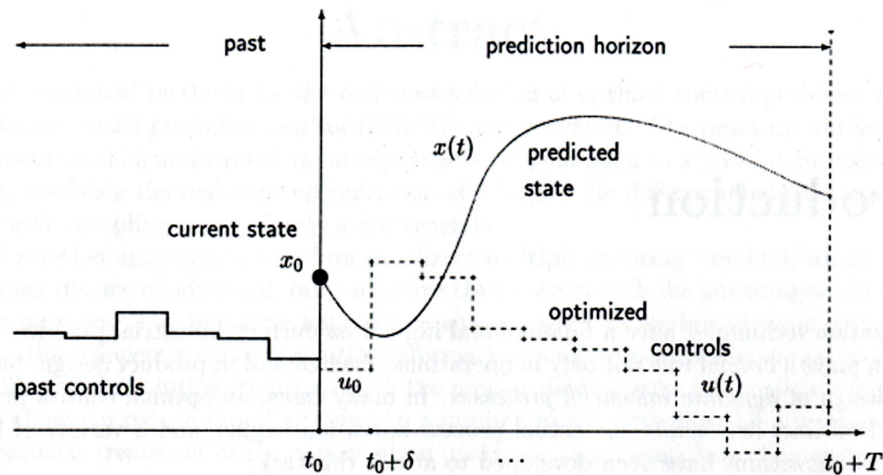


Figure 4 cf.[3]

Néanmoins, dans l'industrie, les NMPC sont encore perçus comme un outil académique plutôt qu'une stratégie de commande applicable. Cela vient essentiellement du fait qu'il est difficile de résoudre un problème de commande optimale non linéaire en temps réel.

Je ne détaillerai pas toutes les approches possibles pour surmonter cette difficulté. Le schéma le plus standard est basé sur l'idée que les problèmes d'optimisation à horizon glissant peuvent être formulés les uns à la suite des autres et que chacun de ces problèmes peut être résolu de manière indépendante avec une précision choisie. La méthode classique pour cela est typiquement une méthode SQP (*Sequential Quadratic Programming*) qui consiste à faire des approximations quadratiques du critère et linéaires des contraintes pour résoudre un problème d'optimisation non linéaire. On peut résumer ce schéma "classique" de la façon suivante:

1. Formulation du problème d'optimisation avec les données de l'étape k .
 - (a) Initialisation de la méthode de calcul de la solution
 - (b) Itérations

- (c) Arrêt quand un critère d'arrêt est vérifié (ou quand un temps limite est atteint)
- 2. On donne la première valeur du contrôle à l'usine
- 3. On passe de k à $k + 1$ et on retourne à 1.

Il faut choisir une méthode off-line très efficace et formuler les problèmes d'optimisation de telle sorte que la contrainte "temps réel" soit respectée. On peut aussi remarquer qu'un retard d'un échantillon de temps apparaît dans ce schéma.

5.2 Itérations en temps réel dans un modèle de commande prédictive

Récemment, plusieurs travaux ont apporté des améliorations à ce type de démarche. Je citerai tout d'abord le schéma d'itérations en temps réel, dû à une équipe de mathématiciens et d'automaticiens de Heidelberg (cf [3])

Le schéma peut se résumer de la façon suivante :

1. Préparation de la k ième itération en temps réel autant que possible sans la connaissance des données de l'étape k .
2. Quand les données de l'étape k sont disponibles, on modifie le problème, et on fait les calculs qui permettent d'obtenir la première valeur du contrôle rapidement.
3. On donne cette valeur immédiatement à l'usine
4. On finit les calculs de la k ième itération.
5. On passe de k à $k + 1$ et on retourne à 1.

Cette approche ne fait qu'une seule itération par pas de temps et autorise ainsi à réduire les pas de temps considérablement. De plus, l'étape feedback 2. est elle même bien plus courte qu'une itération complète de sorte que le retard dans la réponse peut être évité. Ce schéma a été testé avec succès pour la commande d'une colonne de distillation.

5.3 Commande prédictive par modèle interne

Une autre approche est d'utiliser un modèle interne que l'on compare à chaque instant au procédé. Je commente ici des travaux dus à Y. Touré et son équipe et je me réfère en particulier à la thèse de P. Dufour [4].

C'est la problématique posée par l'utilisateur du procédé qui doit être traduite en un problème de commande. A partir du cahier des charges, cela permet de spécifier l'objectif qui peut être:

- d'optimiser un critère de performance lié à des critères économiques: augmenter le rendement, diminuer les pertes de matières premières, etc.
- d'assurer la poursuite d'une trajectoire de référence en sortie du procédé.

C'est un problème assez répandu et cela traduit les performances souhaitées du procédé en boucle fermée, en terme de rapidité et de précision.

L'objectif peut aussi correspondre à un comportement optimal vis-à-vis de la qualité du produit à élaborer. Les contraintes de fonctionnement que doivent respecter certaines grandeurs physiques :

- les actionneurs sont généralement limités en amplitude. Ils peuvent l'être aussi en vitesse et en accélération.
- certaines grandeurs du procédé, mesurées ou estimées, doivent rester dans une zone de fonctionnement.

Le problème d'optimisation est la traduction mathématique des objectifs de contrôle du procédé sous les contraintes de fonctionnement. Pour un système continu, toutes ces grandeurs dépendent du temps et de la (ou des) variable(s) spatiale(s). Il est évident qu'une telle forme est difficilement exploitable pour l'étude mathématique et même pour la résolution tout court: c'est un problème d'optimisation en dimension infinie. On ramène alors le problème d'optimisation à un problème plus classique d'optimisation en dimension finie, pour lequel les résultats et les outils existent en grand nombre.

Dans la structure de commande par modèle interne (CMI) [6], la commande est non seulement appliquée au procédé, mais aussi à son modèle:

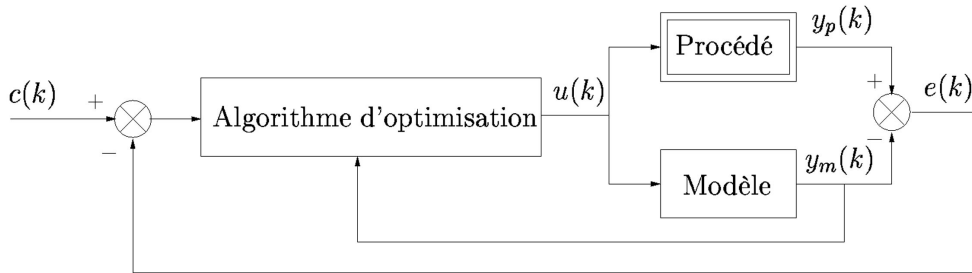


Figure 5 : Structure de Commande Prédictive par Modèle Interne [4]

Le modèle a les caractéristiques suivantes:

- son degré de complexité est lié à la finesse des objectifs à atteindre,
- par rapport au procédé, il doit être qualitativement juste au niveau des tendances:
 - il est dissipatif quand le procédé l'est,
 - il intègre un retard présent dans le procédé,
 - une réaction endothermique n'est pas modélisée de façon exothermique, etc..

En outre, dans une certaine mesure, des erreurs de modélisation et des erreurs additives en sortie de procédé peuvent être acceptées dans cette structure sans que cela ne vienne perturber les résultats.

Les sorties du procédé $y_p(k)$ peuvent donc, d'après la structure représentée dans la figure 4. s'élaborer à partir du modèle et du signal d'erreur $e(k)$:

$$y_p(k) = Y_m(k) + e(k)$$

Cette structure étant valable à tout instant, elle l'est aussi aux instants ultérieurs j :

$$Y_p(j) = Y_m(j) + e(j) \quad (j \geq T)$$

Si le signal de sortie futur $y_p(j)$ peut être évalué à partir du signal de sortie du modèle $Y_m(j)$ résolu numériquement, il reste encore tributaire du signal d'erreur futur $e(j)$. Ce signal étant forcément inconnu à l'instant présent k , il est considéré comme une perturbation de sortie mais assurant tout de même

le bouclage de la structure de commande. C'est donc une information primordiale et se pose donc le problème de sa quantification. Des modèles d'estimation d'écarts ont été donnés par Morari [6]. La forme la plus simple est de considérer l'écart constant sur l'horizon de prédiction, cet écart étant mis à jour à chaque période d'échantillonnage par la nouvelle mesure des sorties du procédé et le dernier calcul du modèle.

En résumé, la problématique issue du cahier des charges se traduit en un problème d'optimisation sous contraintes initialement posé en dimension infinie. Une discrétisation du temps et une autre concernant l'argument de commande permettent de poser ce problème en dimension finie. Cela permet également d'introduire les paramètres de réglage essentiels de la commande prédictive: les horizons de commande et de prédiction.

La structure de commande en boucle fermée est donc basée sur la structure de commande par modèle interne. Celle-ci permet d'intégrer dans le problème d'optimisation sous contraintes en dimension finie non seulement les mesures issues du procédé, mais aussi son comportement futur par le biais du modèle. Cette approche a été décrite pour une problématique générale et pour un problème de poursuite de trajectoire en particulier. Donnons à titre d'exemple un schéma correspondant à la poursuite de trajectoires :

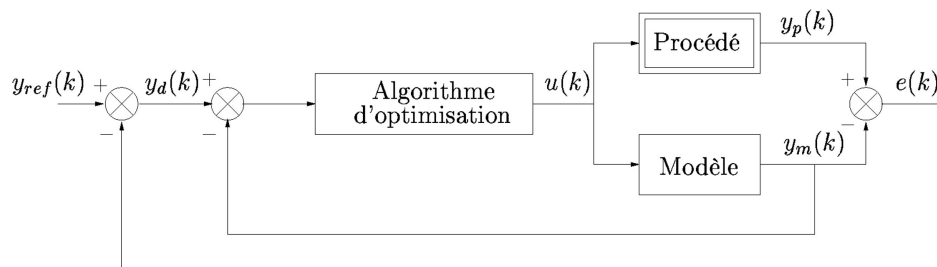


Figure 6 : Structure de commande pour la poursuite de trajectoires [4]

Ces techniques ont été appliquées avec succès pour la commande de la cuisson d'un polymère dans un autoclave et plus récemment pour un procédé de séchage de peinture par infrarouge.

En conclusion, il semble que la commande prédictive (bien adaptée aux problèmes en temps réel) est aussi un terrain où les deux compétences peuvent

se compléter naturellement. Aux automaticiens, le soin de modéliser les procédés, de gérer les modèles et les stratégies de commande (via par exemple l'ajustement des paramètres). Aux mathématiciens, le soin d'améliorer les algorithmes de résolution des problèmes d'optimisation et de faire une étude qualitative de ces solutions.

6 Conclusion

En conclusion, je pense que les deux disciplines ont beaucoup à gagner à se rapprocher. Seules les méthodes et les points de vue diffèrent : les problèmes sont en général les mêmes. Je constate d'ailleurs avec plaisir que les rapprochements sont de plus en plus nombreux, aussi bien au niveau des chercheurs que des étudiants. Je peux ainsi témoigner que des étudiants ayant effectué leur cursus en mathématiques jusqu'à BAC +4, se sont pleinement investis en automatique au niveau DEA/DESS et pour certains ont commencé des thèses en automatique.

Références

- [1] M. Bergounioux. A quel moment peut-t'on discrétiser en contrôle optimal? In *Actes CIFA 2000*, pages 366–368, 2000.
- [2] M. Bergounioux. *Optimisation et contrôle des systèmes linéaires*. Dunod, 2001.
- [3] M. Diehl. *Real -Time Optimization for Large Scale Nonlinear Processes*. PhD thesis, Ruprecht-Karls-Universität, Heidelberg, 2001.
- [4] P. Dufour. *Commande prédictive de systèmes non linéaires à paramètres répartis et applications*. PhD thesis, Université d'Orléans, 2000.
- [5] A. Rey (ed.). *Le Petit Robert*. Société du Nouveau Littré, 1970.
- [6] M.Morari et E. Zafiriou. *Robust Control*. Dunod, 1983.
- [7] C.E. Garcia D.M. Prett et M. Morari. Model predictive control: Theory and practice- a survey. *Automatica*, page 25:335, 1989.

- [8] C. Gasquet et P. Wittomski. *Analyse de Fourier et Applications*. Masson, Paris, 1997.
- [9] S.J. Qin et T.A. Badgwell. An overview of industrial model predictive control technology. In *Fifth International Conference of Chemical Process Control - CPC V*, page 232:256. American Institute of Chemical Engineers, 1996.
- [10] J-L. Lions. *Contrôle optimal des équations aux dérivées partielles*. Dunod, Paris, 1968.
- [11] P. Rouchon. Contrôle de systèmes décrits par des équations aux dérivées partielles : deux exemples types. In *Actes CIFA 2000*, pages 988–1000, 2000.