

ALASCA — Architecture logicielles avancées pour les systèmes cyber-physiques autonomiques

© Jacques Malenfant

Master informatique, spécialité STL – UFR 919 Ingénierie

Sorbonne Université
Jacques.Malenfant@lip6.fr

Cours 4 Contrôle et autonomicité

Objectifs pédagogiques du cours 4

- Développer les notions fondamentales du **contrôle en boucle fermée** et comprendre à travers un exemple de l'informatique autonome le lien entre la théorie du contrôle et l'auto-adaptabilité.
- Introduire quatre **visions** du contrôle des systèmes :
 - celle de l'*automaticien* et des systèmes le plus souvent *déterministes* et *continus*,
 - celle de l'*informaticien* et des systèmes *discrets* le plus souvent *déterministes* également,
 - celle du spécialiste de la *décision* et des systèmes le plus souvent *stochastiques* et
 - celle du spécialiste des *systèmes hybrides*, à la fois *discrets* et *continus* mais le plus souvent déterministes.
- Comprendre ces visions de manière suffisante pour identifier le type de modèles à utiliser en pratique et pour **échanger avec les spécialistes** appropriés pour définir un modèle et une solution corrects.
- Par une vision de la théorie du contrôle, identifier les principales problématiques de la **conception** et la **mise au point** des systèmes auto-adaptables qui seront couvertes dans la suite de l'UE.

Systèmes cyber-physiques autonomiques et contrôle

- Au cœur des systèmes de contrôle cyber-physiques se trouve le contrôle en boucle fermée venant de l'automatique.
- C'est aussi l'inspiration fondamentale de l'informatique autonome.
- Approche *ad hoc* : quand peut-on ignorer les problématiques connues en contrôle ?
 - quand les adaptations sont relativement *rare*s, bien *isolées* les unes des autres, de *coût* et *durée* faibles et
 - guidées par des évolutions de l'entité contrôlée ou de l'environnement qui sont *lentes* et *prédictibles* en comparaison de la durée et du coût des adaptations.
- Par contre, dès que
 - les conditions évoluent *rapidement* d'une manière *imprévisible* et partiellement *indépendante* de l'exécution du système, et
 - nécessitent des adaptations dont le *coût* et la *durée* sont *sensibles* par rapport à leur fréquence potentielle,
 il est *crucial* de comprendre les enseignements de la théorie du contrôle pour mettre en œuvre des systèmes *viables*.

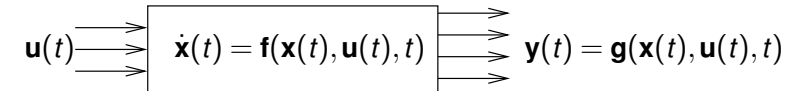
- 1 Introduction à la théorie du contrôle
- 2 Problématique du contrôle
- 3 L'art et la science de l'automaticien
- 4 Contrôle des systèmes hybrides

- En régulation, le contrôle vise à maintenir la sortie mesurée au plus près d'une référence à préserver au cours du temps.
Comment exprimer cette référence ?
- On appelle *signal de référence* une fonction $r(t)$ prescrivant le comportement du système, c'est-à-dire la trajectoire que l'on souhaite lui voir suivre en fonction du temps.
- Ce signal de référence peut servir au contrôleur à calculer les valeurs des variables d'entrée u qui permettent d'influencer le comportement du système :

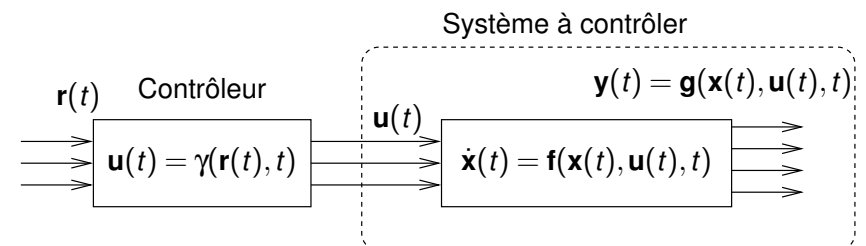
$$u(t) = \gamma(r(t), t)$$

- Nota : en théorie du contrôle, γ s'appellent une *loi ou politique de contrôle*.

- Nous avons vu qu'un système continu peut être modélisé par des équations d'états résumées par le schéma suivant :



- Dans ce schéma, la fonction $u(t)$ est une *entrée contrôlée* qui influence la valeur de la *sortie* $y(t)$ à *réguler*.
 - La théorie du contrôle ayant été d'abord développée pour réguler des procédés mécaniques, électriques, chimiques, etc., son vocabulaire est souvent orienté vers ce type de procédés.
 - Pour l'appliquer en informatique autonome, il faut les prendre toujours dans une acception assez large.
- Grande question de la théorie du contrôle : comment déterminer $u(t)$ pour obtenir un $y(t)$ ayant de *bonnes* propriétés ?



Rappel :

- $r(t)$: signal de référence
- γ : loi de contrôle

Pour mieux comprendre, prenons un exemple concret :

- Considérons le pilotage d'une voiture sur un certain trajet.
- La référence $r(t)$ serait alors la description (mathématique) du trajet c'est-à-dire, par exemple,
 - les positions GPS successives en fonction du temps t ,
 - mais on peut aussi ajouter d'autres éléments comme le temps alloué par étape ou la vitesse maximale en tout temps t , etc.
- Si on suppose, en simplifiant, que les variables d'entrée comportent :
 - un angle α imprimé au volant pour obtenir une direction et
 - une pression p à exercer sur l'accélérateur pour obtenir une vitesse v ,

la fonction γ calculerait à tout instant t l'angle du volant $\alpha(t)$ et la pression sur l'accélérateur $p(t)$ permettant de suivre la trajectoire $y(t)$ aussi près que possible de $r(t)$ en respectant toutes les contraintes données.

- L'utilisation du principe de rétroaction consiste à permettre d'utiliser l'état $x(t)$ du système au temps t dans le modèle pour calculer les nouvelles entrées, ce qui donne une loi de commande de la forme¹ :

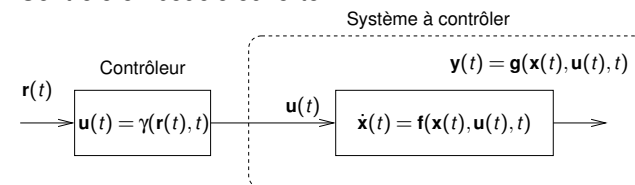
$$u(t) = \gamma(r(t), x(t), t)$$

- Connaître l'état permet de calculer l'écart entre l'état courant et la référence, et ainsi produire de nouvelles entrées susceptibles de *corriger* cet écart.
- On parle alors de contrôle en boucle fermée, et par opposition un contrôle *n'utilisant pas* l'état courant du système sera appelé un contrôle en boucle ouverte.

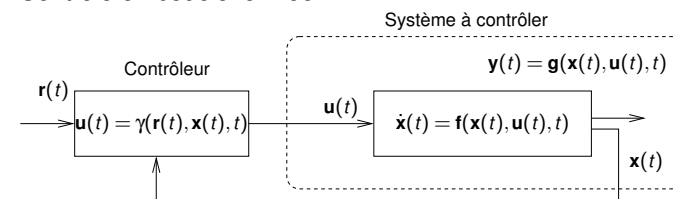
¹ Une formulation alternative utilise une boucle sur la sortie $y(t)$ i.e., une formulation $u(t) = \gamma(r(t), y(t), t)$. Les deux sont donc possibles.

- Bien sûr, conduire ne se résume pas à tourner le volant et appuyer sur l'accélérateur de manière prédéterminée !
- En théorie, certes, si tout est connu et déterministe, la position à tout instant t serait précalculable de manière sûre à partir d'une position initiale ainsi que des angles et des vitesses à tout instant.
 ⇒ Le contrôle devient un problème d'*inversion* i.e., obtenir les angles et vitesses à partir d'une trajectoire souhaitée.
- Malheureusement, le système subit des perturbations : vent, état de la chaussée, piétons, autres véhicules, etc.
- Naturellement, un conducteur utilise le retour visuel (et parfois auditif ; -) qu'il perçoit sur sa trajectoire et de son environnement pour corriger sa vitesse et son angle de direction.
- *Ce retour visuel réalise une **rétroaction**, à savoir la connaissance de l'état courant du système, pour calculer les nouvelles valeurs des variables d'entrée.*

Contrôle en boucle ouverte :



Contrôle en boucle fermée :



Pourquoi une « théorie du contrôle » ?

- En principe, obtenir la loi de contrôle γ paraît relativement simple :
Étant donnée une référence $r(t)$ et l'état courant $x(t)$, il suffit que γ calcule l'écart et connaissant les équations d'états, imprime aux variables d'entrée les valeurs nécessaires pour corriger tout cet écart et se ramener à la référence.
- Les problèmes soulevés par l'obtention de γ sont nombreux :
 - les équations d'états peuvent être difficiles à inverser pour obtenir les valeurs d'entrées corrigeant un certain écart ;
 - mais surtout, ce sont les *contraintes* imposées et le *temps* qui sont les sources principales de difficultés : *atteint-on les limites des actions possibles ? peut-on corriger à temps ?*
- Trouver des lois de contrôles correctes en fonction des contraintes pour des systèmes soumis à des perturbations parfois aléatoires, voilà toute la complexité de la théorie du contrôle !
- Bien que très actif surtout depuis environ 75 ans, le calcul des lois de contrôle optimales pour tous types de systèmes est une thématique de recherche loin d'être épuisée...

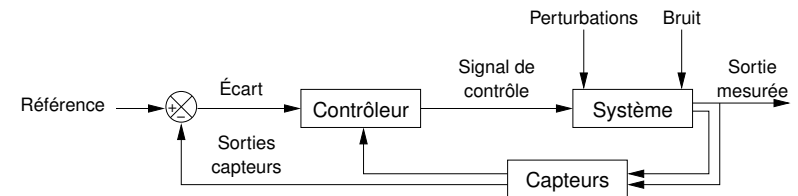
Retour sur l'exemple de la conduite

- Si on considère à nouveau l'exemple de la conduite automobile, des contraintes apparaissent rapidement :
 - l'angle de direction en fonction de la vitesse (on ne braque pas à 90° à 130 km/h...),
 - la rapidité avec laquelle le véhicule peut réduire sa vitesse (on ne passe pas instantanément de 130 à 50 km/h),
 - vitesse maximale de la voiture à un instant donné (on ne roule pas à 250 km/h, enfin... ;-), etc.
- Exemples de contrôles n'atteignant pas leurs objectifs :
 - Si l'angle de direction α est borné par $\pm \alpha_{max}^v$ pour une certaine vitesse v , une correction de trajectoire *souhaitée* peut devenir *impossible* si le délai de réaction est trop court.
 - Si on a pris trop de retard par rapport au trajet, les limitations de vitesse peuvent empêcher de compenser ce retard, et donc la position réelle de la voiture va rester durablement en arrière de la position souhaitée par la référence.

Plan

- Introduction à la théorie du contrôle
- Problématique du contrôle
- L'art et la science de l'automaticien
- Contrôle des systèmes hybrides

Schéma complet d'un système contrôlé en boucle fermée



- Système :** le système contrôlé.
- Perturbations :** toute influence sur la manière dont l'entrée affecte le système de manière non-contrôlée *i.e.*, paramètres externes non-contrôlés.
- Bruit :** toute influence sur la qualité de la mesure de la sortie.
- Sortie mesurée :** valeurs des propriétés mesurables du système.
- Capteurs :** dispositifs de mesure de l'état et la sortie courants.
- Référence :** valeurs souhaitées pour la sortie mesurable.
- Écart :** différences entre les valeurs de référence et de la sortie capteurs.
- Contrôleur :** calcule les valeurs de l'entrée contrôlée à partir des écarts.
- Signal de contrôle :** valeurs de l'entrée contrôlée.

Types d'objectifs

Les objectifs d'un contrôle en boucle fermée peuvent être de trois ordres :

Régulation : s'assure que la trajectoire de l'état demeure autant que possible égale ou le plus près possible de la référence.

Élimination des perturbations : s'assure que les perturbations n'affectent pas significativement la sortie mesurée en agissant contre ces perturbations.

Optimisation : s'assure que la sortie mesurée est la meilleure possible étant donné un certain critère (coût) exprimé en fonction de l'état et/ou de la sortie mesurée.

Types d'approches de résolution I

- La problématique du contrôle est apparue dans différents *domaines* selon différentes *approches* et avec différentes *préoccupations*, parmi lesquelles on peut identifier :
 - L'approche de l'*automatique*, construite essentiellement autour des systèmes physiques déterministes et continus, s'est principalement intéressée à leur *stabilisation* rapide et efficace.
 - L'approche de l'*informatique concurrente*, fondée sur les automates, s'est principalement intéressée à vérifier les propriétés comme l'*atteignabilité* des états et la capacité à amener le système dans certains états en forçant ou inhibant certaines actions de contrôle (par exemple, pour éviter l'étreinte fatale).
 - L'approche des processus markoviens commandés de la *décision*, qui s'est principalement intéressée à trouver des *lois de contrôle optimales* selon des critères donnés, en présence d'aléas dans l'évolution du système et les effets des actions de contrôle.

Types d'approches de résolution II

- L'approche des *systèmes hybrides* qui souhaite attaquer des systèmes où co-existent les propriétés des deux précédentes approches avec les perturbations engendrées par les sauts et les commutations sur la qualité des contrôles.
- Les différents outils théoriques et pratiques ayant été développés dans des contextes différents, leurs hypothèses sont aussi différentes mais complémentaires :
 - Savoir parler à chacun des groupes de spécialistes en comprenant son angle d'attaque des problèmes est crucial.
 - Savoir identifier le type de système à contrôler et utiliser les bonnes approches est important.
 - Savoir quand il faut combiner certaines approches pour résoudre les problèmes plus complexes est appréciable.

Exemple : allocation dynamique de machines virtuelles I

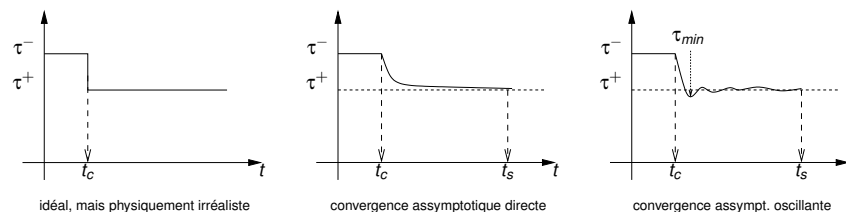
- Application web s'exécutant sur une ferme de serveurs utilisant des machines virtuelles (MV).
- Élasticité :
 - **verticale** : modifier la quantité de ressources allouée aux MV pour augmenter leur capacité de traitement ;
 - **horizontale** : modifier le nombre de MV allouées à l'application.
- Objectif : faire varier le nombre ou la taille des MV de manière à maintenir τ , le temps moyen d'exécution des requêtes (attente + service), autour d'une valeur prédéfinie τ_r .
 - Si τ diminue, le coût d'opération augmente inutilement, et donc on peut réduire le nombre ou la taille des MV.
 - Si τ augmente, les clients fuient, donc il faut augmenter le nombre ou la taille des MV.

Exemple : allocation dynamique de machines virtuelles II

- En termes du schéma de système contrôlé en boucle fermée :
 - Système** : application et ses MV.
 - État et sortie mesurée** : nombre de MV courant et τ .
 - Capteurs** : mesure et échantillonnage de τ donnant une sortie capteur $\hat{\tau}$.
 - Référence** : τ_r .
 - Contrôleur** : entité qui décide quand et combien de MV il faut ajouter ou retrancher en fonction de l'écart entre la sortie capteur $\hat{\tau}$ et la référence τ_r .
 - Signal de contrôle** : ordre d'ajout ou de retrait de MV.
 - Perturbations** : variation de τ_a , le taux d'arrivée des requêtes et de la répartition des types de requêtes venant des clients.
 - Bruit** : erreurs de mesure de $\hat{\tau}$.

Identification du système

- Quel est le τ en fonction de τ_a et du nombre de MV (de la taille choisie) ? En fait M/M/1 i.e., τ_a stable $\Rightarrow \tau$ se stabilise.
Exemple : on peut faire des expériences et mesurer. Supposons qu'on obtient que pour $\tau_a = 400$ r/s, une MV maintient $\tau = 1$ s et que les données expérimentales montrent que le τ à stabilisation pour un τ_a constant varie linéairement en fonction de τ_a et du nombre de MV (i.e., règle de 3).
- Comment le système réagit-il lors de l'ajout et du retrait de MV ?
Exemple : on a 2 MV et $\tau_a = 1000$ r/s, donc $\tau^- = 1.25$ s ; au temps t_c on ajoute une MV, à τ_a constant, donc on devrait arriver à $\tau^+ = 0.833$ s, mais comment ?



Et quel serait l'effet de la variation de τ_a pendant ce temps (perturbation) ?

Le contrôle en pratique

- Pourquoi un contrôle comme la régulation du temps moyen d'exécution des requêtes pose-t-il tant de problèmes ?
- Ils viennent essentiellement du fait :
 - que l'effet d'une adaptation n'est pas toujours *immédiat* ;
 - que cet effet peut induire un *comportement transitoire indésirable*, comme des oscillations dans la valeur (sortie) mesurée ;
 - que l'effet des adaptations possibles peut être *trop puissant* ou au contraire *insuffisant* à cause des limites intrinsèques des ces adaptations ;
 - que les *perturbations* peuvent être aussi fortes que l'on veut et donc *dépasser les capacités* des adaptations ;
 - qu'un *bruit trop important* rende la valeur mesurée inutilisable.
- Examinons cela sur l'exemple pour mieux comprendre.

Problèmes qui peuvent surgir dans le choix de l'adaptation I

Supposons qu'on choisit un délai Δ entre les adaptations d'au moins $\Delta > t_s - t_c$, i.e. $t_{i+1} = t_i + \Delta$ (sinon, instabilité...).

Scénario 1 : à chaque t_i , +1 MV si $\tau > 1.2$ s et -1 MV $\tau < 0.9$ s.

- État : $\tau_a = 1000$ r/s, 2 MV, $\tau = 1.25$ s.
- Évolution prévisible à τ_a constant :
 - t_i : État : 2 MV, $\tau = 1.25$, Décision : +1 MV.
 - t_{i+1} : État : 3 MV, $\tau = 0.833$, Décision : -1 MV.
 - t_{i+2} : État : 2 MV, $\tau = 1.25$, Décision : +1 MV.
 - t_{i+3} : État : 3 MV, $\tau = 0.833$, Décision : -1 MV.
 - ...
- τ oscille entre 0.833 s et 1.25 s en ajoutant ou retranchant une MV à chaque instant de contrôle $\Rightarrow$ action trop puissante !
- Et le client paie à chaque fois le coût fixe d'allocation...

Petit coup d'oeil sur la suite du cours... I

- Les difficultés induites par le **coût des mesures** et les **erreurs** dont elles peuvent être entachées sur l'état seront discutées dans au cours 7.
- Dans cet exemple, l'ajout d'une machine virtuelle peut demander de la créer, d'où un coût et un délai de mise à disposition ; plus généralement les difficultés induites par les **actionneurs** seront discutées un peu aujourd'hui et mais aussi dans les cours 7 et 8.
- Une fois qu'une MV a été ajoutée, il faudra du temps avant que le temps moyen d'exécution des requêtes se stabilise à son nouveau niveau (à taux d'arrivées des requêtes constant) ; ces difficultés liées à l'**inertie du système** sont discutées dans le reste du cours d'aujourd'hui et aux cours 11 et 13.

Petit coup d'oeil sur la suite du cours... II

- Le choix du nombre de MV à ajouter ou retrancher doit être fait judicieusement pour éviter d'augmenter le coût d'opération : c'est la politique de contrôle ; les difficultés liées au calcul d'une **bonne politique de contrôle** seront couvertes dans le reste du cours aujourd'hui et au cours 13. Il en va de même pour l'effet de la variation des perturbations, comme le taux d'arrivées des requêtes dans notre exemple.
- La théorie du contrôle examine un système à partir d'un modèle, mais en pratique il faut implanter ce modèle sur le système réel avec des entités logicielles qu'on voudra développer de manière efficace et réutiliser dans toute la mesure du possible ; les difficultés liées au développement et au **génie logiciel** appliqués aux entités logicielles des systèmes autonomiques seront couvertes dans les cours 8, 10 et 11.

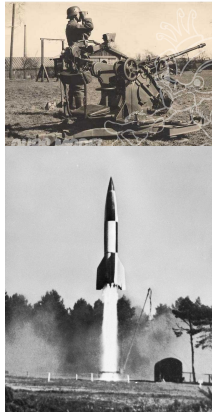
Petit coup d'oeil sur la suite du cours... III

- Une ferme de serveurs exécute rarement une seule application, or l'allocation de ses ressources aux différentes applications nécessitent une **coordination** entre chacune ; les difficultés liées à la coordination entre les entités autonomiques seront couvertes aux cours 12, et leur impact sur le développement des entités seront introduites aux cours 8, 10 et 11.
- Si le système évolue ou ses perturbations et bruits changent, sa loin de commande devra aussi évoluer, ce qui donne un contrôle adaptatif qui doit lui-même être régulé ; cet aspect de contrôle du contrôle ou **méta-contrôle** sera discuté au cours 13.
- Le **passage à l'échelle** sera abordé en coordination au cours 12 et dans les systèmes autonomiques en général au cours 14.

Plan

- 1 Introduction à la théorie du contrôle
- 2 Problématique du contrôle
- 3 L'art et la science de l'automaticien**
- 4 Contrôle des systèmes hybrides

- L'automatique est née du besoin de contrôler des phénomènes essentiellement *analogiques* (continus, non numérisés) à l'aide de dispositifs eux-même *analogiques*.
- Les techniques employées consistaient à mesurer l'écart analogiquement, par un voltage par exemple, puis d'amplifier et transformer ce voltage pour impulser une action sur un actionneur.
- Exemples :
 - Artillerie anti-aérienne : régler la direction de tir (vérins, moteurs) en fonction de la position de la cible.
 - Missile balistique : suivre une trajectoire en agissant sur l'angle du jet de gaz du moteur et des gouvernes aérodynamiques des ailerons.



- C'est l'*inertie* dans le temps propre à ces phénomènes, liées aux limites physiques intrinsèques des actionneurs (souvent mécaniques, *in fine*), qui a justifié une grande part des travaux en automatique (ou théorie du contrôle).
 - Sachant qu'il s'agit d'influer sur une propriété mesurée à partir d'une propriété contrôlée, la grande question est :
comment obtenir une « trajectoire » de référence pour un phénomène évoluant selon sa dynamique propre (trajectoire intrinsèque) en lui appliquant une correction en continu ?
- et subsidiairement, comment obtenir une trajectoire de *bonne qualité* (i.e., SASO) ou optimale ?
- Pour attaquer formellement ce problème, les automaticiens se sont d'abord appliqués à comprendre les signaux et les effets de leur *combinaison*.

- Les phénomènes étudiés en automatique sont des propriétés dont la valeur évolue dans le temps.
- Soit une propriété p , qui prend à tout instant t une certaine valeur, il est naturel de voir p comme une fonction du temps $p(t)$.
 ⇒ En automatique, on appelle cela un *signal*.
- Contrôler consiste à appliquer un signal de contrôle, noté bien sûr $u(t)$, au signal $p(t)$ représentant la propriété à contrôler.
- L'effet de $u(t)$ est de se combiner à $p(t)$ de manière à en changer l'évolution, ce qui peut être capturé par une *équation de récurrence* dont la forme générale est :

$$p(t) = f(p(t), u(t))$$
- Avant même de se soucier du problème consistant à déterminer quel contrôle $u(t)$ appliquer à un signal $p(t)$ pour atteindre un certain objectif, l'automaticien étudie les propriétés de base des combinaisons $f(\cdot, \cdot)$ de signaux $p(t)$ et $u(t)$ connus.

- Lorsqu'on combine des valeurs de signaux, les effets peuvent être de plusieurs ordres : additifs (ou linéaires), multiplicatifs (quadratiques, etc.), ou plus généralement non-linéaires.
- L'étude théorique des propriétés des combinaisons de signaux est un domaine très complexe en général, donc pour des raisons pratiques tenant à la fois aux possibilités de la théorie et à la réalité des applications de l'automatique, une grande part des études ont porté sur les systèmes *linéaires* où :

$$p(t) = f(p(t), u(t)) = ap(t) + bu(t)$$
- Pour des raisons historiques également, vu les premières applications, l'étude s'est aussi concentrée sur des systèmes *déterministes*, c'est-à-dire où les erreurs et autres effets aléatoires sont soit *inexistants*, soit *négligeables* (au sens de pouvant être négligés sans trop de problème).

Le prototype de l'automatique linéaire déterministe

- L'équation $p(t) = ap(t) + bu(t)$ est une *combinaison linéaire récurrente* dont il faut trouver une solution *i.e.*, deux alternatives :
 - Étant donné $u(t)$, quels a, b tel que $p(t) = r(t)$?
 - Étant donné a, b , quel $u(t)$ tel que $p(t) = r(t)$?
- Dans le cas où le contrôle est effectué par des contrôleurs numériques, les interventions sur le système ne peuvent se faire qu'à des instants discrets, ce qui justifie d'étudier la version discrète de l'équation :

$$p(t_{k+1}) = ap(t_k) + bu(t_k)$$

ou de manière plus simple ($t_{k+1} - t_k$ étant constant) :

$$p(k+1) = ap(k) + bu(k)$$

D'ailleurs, on peut montrer qu'il vaut mieux discrétiser l'équation avant sa résolution que discrétiser une solution continue.

Identification du système

- Le premier problème consiste à *identifier* le système c'est-à-dire obtenir les valeurs des paramètres faisant coller le modèle général au système.
 - ⇒ Dans notre exemple, les valeurs de a et b .
- Pour certains systèmes (électricité, mécanique, etc.), il arrive que les lois connues de la physique permettent de déterminer les équations du modèle.
- Pour de nombreux systèmes, cette approche analytique est impossible car la nature du système est inconnue ou trop complexe.
- On peut alors faire un modèle *boîte noire* avec des méthodes statistiques comme la régression linéaire ou les séries chronologiques à partir de *données expérimentales* mesurées sur un exemplaire du système.
- Par exemple, après expérimentation, on peut obtenir :

$$p(k+1) = 0.43p(k) + 0.47u(k)$$

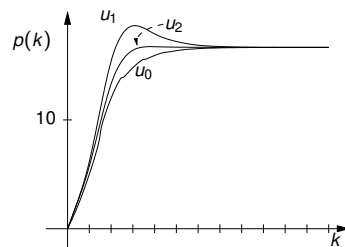
Comment réagit ce signal à un certain contrôle ?

- À partir du modèle (équation) précédent, on peut étudier le comportement du signal $p(t)$ *étant donné* un signal $u(t)$.
- Signaux de contrôle communs :
 - impulsion : $u(0) = U, u(k) = 0, k \geq 1$
 - saut : $u(0) = U_0, u(k) = U, k \geq 1$
 - linéaire : $u(k) = kU, k \geq 0$
- Appliquons à notre modèle trois signaux simples avec comme condition initiale $p(0) = 0$:

$$u_0(k) = \begin{cases} 10 & k=0 \\ 20 & k \geq 1 \end{cases}$$

$$u_1(k) = \begin{cases} 10 & k=0 \\ 30 & k=1 \\ 25 & k=2 \\ 20 & k \geq 3 \end{cases}$$

$$u_2(k) = \begin{cases} 10 & k=0 \\ 25 & k=1 \\ 22 & k=2 \\ 20 & k \geq 3 \end{cases}$$



Analyse des propriétés du modèle

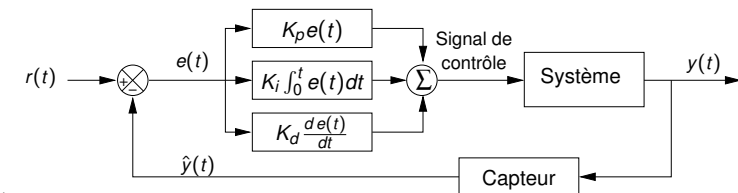
- L'automatique, puisant dans les techniques mathématiques comme les transformées de Laplace et les transformées en Z, a étudié avec beaucoup de soin les propriétés des modèles de combinaison de signaux.
- Selon les propriétés des équations, on peut arriver à conclure des choses comme, pour un $u(k)$ borné ou non :
 - le signal $p(k)$ est stable ou instable ;
 - le signal $p(k)$ oscille ou non, et s'il oscille, la fréquence de ces oscillations ;
 - le signal $p(k)$ converge à ε près en moins de n étapes.
- Et donc d'analyser dans ces conditions les propriétés SASO de ce système.

De l'analyse au choix du contrôle $u(t)$

- Dans les analyses précédentes, on s'intéresse aux propriétés générales du signal de sortie pour tout signal de contrôle.
- Ce petit énoncé introductif néglige beaucoup de choses, dont les effets des perturbations et du bruit : *on y reviendra* !
- La question brûlante est comment choisir le contrôle $u(k)$?
 - en régulation, il faut réduire l'écart entre l'état $p(k)$ et la référence $r(k)$, il est donc raisonnable de construire $u(k)$ comme fonction de $e(k)$, l'écart $p(k) - r(k)$ vu comme un signal ;
 - en élimination des perturbations, c'est le signal de perturbations qu'on veut éliminer et donc on peut construire le contrôle en fonction de ce que l'on connaît de lui, déjà plus compliqué... ;
 - en optimisation, c'est encore plus complexe car on va choisir un signal de contrôle qui va minimiser une certaine fonction de coût C , et généralement le coût total qu'elle induit $\sum_0^N C(p(k), u(k))$.

Un contrôleur linéaire très courant : PID

- Considérons un contrôle en régulation où $u(k+1) = g(e(k))$ ($e(k)$ étant l'erreur mesurée). Mais *quelle* fonction g ?
- Encore une fois, la simplicité et l'efficacité pratique des approches linéaires ont amené à étudier des fonctions g linéaires.
- Poser $u(k+1) = K_p e(k)$, c'est-à-dire *proportionnel* (P) à l'erreur, est cependant un peu simple et donne des résultats décevants.
- Une famille de contrôleurs qui se sont montrés efficaces en pratique et donc sont beaucoup utilisés est celle des contrôleurs *PID*, pour *proportionnel*, *intégrale*¹ et *dérivée*.



¹ Tend vers 0, pour un contrôle efficace où

l'erreur se répartit en moyenne également autour de 0.

Détermination des constantes de proportionalité I

- En discrétisant, on obtient un contrôle de la forme :

$$u(k+1) = K_p e(k) + K_i \sum_{j=0}^k e(j) + K_d (e(k) - e(k-1))$$

- il reste à déterminer les constantes K_p , K_i et K_d
- Encore une fois, les outils mathématiques développés en automatique permettent de dégrossir les choses : en régulation, un contrôleur PID réussira à éliminer l'erreur, mais comment $y(t)$ va-t-elle évoluer ?
- L'automaticien va déterminer ses critères de sélection :
 - précision et temps maximal de réglage autorisé,
 - oscillations admissibles ou non,
 - maximum de dépassement, etc.

Détermination des constantes de proportionalité II

- En pratique, l'ingénieur automaticien va souvent tracer les courbes des équations obtenues avec différents paramètres K_p , K_i et K_d (en utilisant un outil comme Matlab, Mathematica, Scilab, Octave, etc.), puis examiner les résultats pour choisir le jeu de ces paramètres dont la courbe satisfera au mieux ses critères de sélection.
- On peut aussi utiliser un contrôleur PID en optimisation :
 - On définit la fonction de coût dépendant de $u(k)$ et $p(k)$, plus éventuellement d'autres paramètres internes.
 - On optimise la fonction de coût pour trouver les paramètres K_p , K_i et K_d tels qu'elle est optimale.

- Comme nous l'avons fait remarquer, les équations précédentes ne font pas apparaître les notions de perturbation et de bruit.
 - Cela revient à considérer que leurs effets, plus ou moins constants, plus ou moins négligeables, peuvent être complètement capturés par les paramètres de l'équation obtenus après modélisation statistique sur données expérimentales.
 - Il est possible de faire les choses de manière plus explicite.
- Pour le bruit, et les autres problèmes de mesure :
 - on peut modéliser la transformation de $y(t)$ par l'application d'une combinaison avec un signal de bruit explicite ;
 - en postulant une forme à ce signal, on peut l'entrer dans le modèle et en tenir compte pour le calcul du contrôle $u(t)$.
- Pour la perturbation :
 - on peut également introduire dans les équations le signal de perturbation attendu et en tenir compte dans le calcul du contrôle ;
 - il existe également des modèles de contrôle prédictif qui tiennent compte de la forme attendue de la perturbation dans le contrôle.

- La forme de modélisation présentée précédemment est axée sur le principe de *fonction de transfert*.
- Dans cette forme de modélisation, le système n'est vu que comme une entité transformant un signal de contrôle en entrée vers un signal mesuré en sortie, laissant sa dynamique interne masquée à l'intérieur d'une boîte noire (sans état explicite).
- Dans nos descriptions précédentes des systèmes, des modèles plus explicites définissent plus finement la dynamique du système au prix d'une plus grande complexité, c'est-à-dire des équations de récurrence incluant l'état x :

$$\begin{aligned} x(k+1) &= f(x(k), u(k), k) & x(k_0) &= x_0 \\ y(k+1) &= g(x(k), u(k), k) \end{aligned}$$

- Parce qu'il est souvent difficile de les expliciter dans leurs applications concrètes, les automaticiens se sont moins intéressés aux modèles à base d'états qu'aux modèles fondés sur les fonctions de transfert simples.
- En informatique cyber-physique et autonome, l'accès à une description des entités contrôlés rend au contraire plus attractifs les modèles à base d'états, mais qui exigent des contrôleurs plus complexes à mettre au point.

- À partir du moment où la trajectoire de l'état est modélisé plus explicitement (et pas seulement celle de la sortie mesurée), il devient intéressant de regarder les propriétés de la dynamique du système entre les contrôles et l'espace d'états.
- Deux propriétés sont principalement considérées :

Contrôlabilité : un système est *contrôlable* si pour tout état final réalisable x_f et tout état x , il existe une séquence de contrôles $[u(0), \dots, u(M)]$ qui, appliquée dans l'état $x(0) = x$, va mener à l'état $x(M) = x_f$.

Observabilité : un système est *observable* si toutes ses valeurs d'états peuvent être déduites uniquement à partir des propriétés mesurées (sorties capteurs, information partielle).

- 1 Introduction à la théorie du contrôle
- 2 Problématique du contrôle
- 3 L'art et la science de l'automaticien
- 4 Contrôle des systèmes hybrides

- Nous avons vu qu'en automatique, l'évolution des systèmes contrôlés, souvent complexe, est étudiée grâce à des systèmes d'équations différentielles ou aux différences où le contrôle est exercé en continu.
- Il est possible de discrétiser les systèmes contrôlés continus, de manière à retomber sur un formalisme discret, mais il ne faut pas confondre :
 - discrétisation par intervalles réguliers des systèmes continus, et
 - systèmes à événements discrets en temps continu, où les transitions sont observées à des instants ponctuels, certes, mais qui n'obéissent à aucune régularité d'occurrence les uns par rapport aux autres.
- Les systèmes hybrides visent à unifier dans un formalisme unique les comportements par événements par essence discrets et les comportements continus, éventuellement discrétisés.

- Pour un système informatisé où l'application du contrôle est assurée par un programme, la nature discrète des calculs impose une discrétisation de la partie continue.
- La théorie du contrôle nous montre qu'il vaut beaucoup mieux discrétiser un problème et trouver la loi de commande pour le problème discret que de trouver une loi de commande continue pour le problème continu et ensuite discrétiser cette loi.
- Cela dit, la discrétisation des problèmes continus se fait généralement par petits pas de temps dont la durée est beaucoup plus faible que la durée entre deux événements discrets forçant le passage dans un autre état discret.
- Malgré la discrétisation de la partie continue, il demeure des sauts discrets impliquant des changements de loi de commande et ce sont ces passages qui compliquent beaucoup la garantie des propriétés SASO d'un contrôleur hybride.

- 1 L'auto-adaptabilité vue sous l'angle de l'informatique autonome rend les entités logicielles auto-adaptables en leur adjoignant une entité qui réalise un **contrôle en boucle fermée**.
- 2 La **théorie du contrôle** a développé des modèles permettant d'analyser et de trouver des lois de contrôle respectant les propriétés de **stabilité**, de **précision**, de **temps de réglage court** et de **minimum de dépassement** qui s'appliquent bien dans le contexte de l'informatique autonome.
- 3 L'**automaticien** se concentre le plus souvent sur des **modèles continus déterministes** et la **stabilisation** rapide d'un système pendant une action de contrôle.
- 4 Le spécialiste de la **décision** fait apparaître des coûts, cherche le plus souvent à capturer la nature **stochastique** des systèmes réels et applique des approches d'optimisation pour trouver des **lois de contrôle optimales**.
- 5 Le spécialiste des **systèmes hybrides** cherche à **concilier** les visions de l'automaticien et de l'informaticien, tout en appliquant des techniques de l'automaticien ou du spécialiste de la décision pour trouver les bonnes lois de contrôle.

